UNIVERSITY OF ALBERTA

Improvements of Approximation of the Local Mean in the Process of Empirical Mode Decomposition

by

Yao Wang

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Science

in

Engineering Management

Department of Mechanical Engineering

Edmonton, Alberta

Spring 2006

Your file Votre référence
ISBN: 0-494-13905-6
Our file Notre référence
ISBN: 0-494-13905-6

Abstract

In fault detection and assessment, a comprehensive understanding and investigation of advanced signal processing methods is required. The empirical mode decomposition (EMD) method is an adaptive time-frequency domain signal processing method that is completely driven by the data itself. The cubic spline interpolation method has been used to approximate the local mean in the sifting process of EMD, where problems of undershooting and overshooting have been identified.

This study explores approaches to improving the approximation of the local mean to obtain better EMD performance. In this thesis, the modified monotone piecewise Hermite interpolation (MMPHI) method is applied for envelope-mean approximation, because it demonstrates advantages over the cubic spline method. A type of direct approximation of the local mean, i.e. the windowed local mean approach, is also investigated and its merit in identifying impulses is demonstrated. Performance of MMPHI and the windowed local mean approach are also demonstrated when these are used to analyze experimental data obtained from a gearbox.

ACKNOWLEDGEMENTS

I would like to express my thanks to the people who have helped me during my M.Sc. study at the University of Alberta.

First of all, sincere appreciation is given to my supervisor, Dr. Ming J. Zuo, for his support, advice and guidance throughout the course of my graduate program. He has been an excellent model of a researcher who has made seeking truth his academic career. I will always treasure the supervision he has given me.

I would like to thank Dr. Xianfeng Fan and Mr. Zhigang Tian for their precious help over the course of two years. Appreciation also goes to the engineers at Syncrude Research Centre, Mr. Amit Aulakh and Mr. Stuart Corke, for sharing their knowledge with me.

I am so happy to have been working with my colleagues in the Reliability Research Lab: Hui Lin, Siyan Wu, Mulugeta D. Abera, and Wei Li. Discussions with them gave me a new perspective on thinking.

Most importantly, I would like to offer thanks for the unconditional support and love given me by my father, mother and brother; it helped me to bring this work to completion.

I would like to dedicate this work to my girl friend, Ms. Song Tao, whose love continually encourages me to move forward.

CONTENTS

List of Tables

List of Figures

CHAPTER 1

INTRODUCTION

1.1 Introduction

The concepts of signals and systems arise in all areas of technology, ranging from appliances found in homes to very sophisticated engineering devices. In fact, it can be argued that much of the development of high technology is a result of advancements in the theories and techniques of signals and systems. Particularly in the research area of reliability and maintenance, development of condition-based maintenance requires a profound understanding of and investigation into advanced signal processing methods. Reliability has always been an important aspect of the assessment of industrial products and equipment. Good product design is, of course, essential for products with high reliability requirements. No matter how perfect the product design is, however, products deteriorate over time because they are operating under stress or load in the real environment, often involving randomness. Maintenance has thus been introduced as a way of assuring a satisfactory level of reliability during the useful life of a physical asset. The focus of maintenance techniques has changed from breakdown maintenance, to time-based preventive maintenance, to condition-based maintenance (CBM) [38].

Condition-based maintenance is a technique that recommends maintenance actions be based on the information collected through condition monitoring. CBM attempts to avoid unnecessary maintenance activities by taking action only when there is evidence of a physical asset's abnormal behavior. There are three key steps in CBM [33]:

1. Data Acquisition (information collecting)–to obtain data relevant to system conditions;

2. Data Processing (information handling)–to handle and analyze the data or signals collected in step 1 for better understanding and interpretation of the data;

3. Maintenance Decision Making (decision making)–to recommend efficient maintenance policies.

The technologies of signal processing play a very important role in fault detection of machinery since it bridges the gap between collected physical signals and the signatures of faulty conditions. These technologies can be categorized into two types: waveform data analysis and image processing. Time domain analysis and frequency domain analysis are techniques of waveform data analysis that were frequently discussed in past decades. Recently, time-frequency domain analysis has become a focus; its algorithms and technologies have been discussed in hundreds of paper. This thesis studies the empirical mode decomposition (EMD), one time-frequency domain signal processing method, to investigate the possible improvements in the performance of applications.

1.2 Motivation

In time domain analysis, some factors are calculated directly from the time waveform itself to indicate the statistical features of signals. Fourier spectral analysis, which examines the global energy-frequency distribution of signals using the Fourier transform, is the most frequently used form of frequency domain analysis. Unfortunately, the data, whether from physical measurements or numerical modelling, will most likely have one or more of the following problems:

- the total data span is short;
- the data are non-stationary;
- the data represent non-linear processes.

In those cases, time domain or frequency domain analysis alone may be unable to capture meaningful signatures of signals. For example, in a project involving condition-based monitoring of slurry pump wear conditions, signals that were collected from a slurry pump at both brand new and worn-out conditions included not only vibration signals, but also pressure, current, flow rate, and acoustic signals because they were fluctuating as well [56]. Time domain data alone provides little information about the machine's status because the processes of operation are non-stationary and non-linear.

Hence, it is necessary to find an efficient signal-processing method suitable to the data which will make it possible to exploit more information. Empirical mode decomposition (EMD) is a new time-frequency domain signal processing method proposed in [27]. Not only can EMD show features of signals in both the time and frequency domains, but also it is a fully data-driven, self-adaptive

signal processing method that decomposes signals without assuming any basic function or using any pre-determined filter. Although EMD has been proven to be effective and robust in the analysis of non-linear and non-stationary data of many applications, there are still a few areas that need further improvement, as stated in the discussion section of [27]. This thesis focuses on studying improvements of the approximation of local mean in the EMD process in order to obtain better performance as applied to signals.

1.3 Organization of Thesis

The thesis is organized as follows. The motivation for this study is introduced in Chapter 1. The background and relevant literature regarding data processing methods, especially, the empirical mode decomposition, are reviewed in Chapter 2. Chapters 3 and 4 are devoted to improvements in the approximation of the local mean and the verification of these improvements by simulated examples. Finally, conclusions are drawn and future work is suggested in Chapter 5.

Chapter 2

Background and Literature Review

2.1 Review of Data Processing Methods

Using data acquisition systems, raw data sets are collected from devices and stored in computer systems. Pre-processing is then done to eliminate data errors, and, thereby, reduce the possibility that analysis may be ruined by errors. It is the next step, which is data analysis, that is the focus of this thesis. The literature provides a variety of models, algorithms and tools for better comprehension and interpretation of data [29]. The models, algorithms and tools used for data analysis depend mainly on the types of data collected. Condition monitoring data collected from the data acquisition step is versatile. It falls into three categories [29]:

Value type Data collected at a specific time for a condition monitoring variable constitutes a single value. For example, oil analysis data, ambient temperature, atmospheric pressure, and humidity are all value type data in most applications.

Waveform type Data collected at a specific time for a condition monitoring variable constitutes a time series; this is often called a time waveform.

For example, vibration and ultrasonic data are waveform data.

Multi-dimension type Data collected at a specific time epoch for a condition monitoring variable is multi-dimensional. The most common multi-dimensional data is image data such as infrared thermographs, X-ray images, visual images, and so on.

Data processing for waveform and multi-dimension of data is also called signal processing. Various signal processing techniques have been developed to analyze and interpret waveforms to extract useful information for further diagnostic and prognostic purposes.

In condition monitoring, the most common waveform data are vibration signals and acoustic signals. Other waveform data include ultrasonic signals, motor current, flow rate, and so on. In the literature, there are three main categories of waveform data analysis in the field of CBM and fault diagnosis: time-domain analysis, frequency-domain analysis and time-frequency analysis.

2.1.1 Time-domain Analysis

Time-domain analysis is directly based on the time waveform itself. Traditional time-domain analysis calculates characteristic features of time waveform signals as descriptive statistics such as mean, peak, peak-to-peak interval, standard deviation, crest factor, form factor; high order statistics: root mean square (RMS), skewness, kurtosis, etc. In [18], crest factor is defined as the ratio of the crest value (peak value) to the effective value (RMS), and form factor is defined as the ratio of the effective value to the half-period mean value. Thus, the crest factor and the form factor of a sine wave are $\sqrt{2} = 1.414$ and $\pi/(2\sqrt{2}) = 1.111$ respectively. The crest factor is calculated is to give us

a quick idea of how much impact is occurring in a waveform. Impact is often associated with roller bearing wear, cavitation and gear tooth wear [43]. A perfect sine wave contains no impact; therefore crest factors with a value higher than 1.414 imply that there is some degree of impact. The above two factors are combined together to create a revised crest factor (RCF) by [31]. The RCF is obtained by multiplying the original crest factor and the form factor. The average of absolute sample values for one cycle instead of half a cycle are used for the mean value. For a sine wave as a benchmark, the RCF is $\pi/2 = 1.571$.

Kurtosis is defined as the fourth statistical moment, normalized by the standard deviation to the fourth power, which is shown below [43].

$$K = \frac{1}{N\sigma^4}\sum_{i=1}^{N}(x_i - m)^4, \tag{2.1}$$

where N is the total number of data points, σ is the standard deviation, m is the average of the signal, and x_i are the amplitudes of the signal. Kurtosis represents a measure of the flattening of the density probability function near the average value. A well-known value for kurtosis is 3, which is the value of a normal distribution. As a parameter for diagnosing faults in rolling bearings, the values of kurtosis increase with the growth of the defect. That happens because the pulses generated are increased by the passage of the rolling elements over the defect [43]. Kurtosis is also used on the diagnosis of bearing wear faults of pumps used in the waste water industry [53]. The reason for using kurtosis is that vibration from an undamaged bearing is not impulsive whereas vibration from a damaged bearing will be impulsive and will result in a kurtosis value greater than 3.

Lin *et al* [35] proposed the fault growth parameter (FGP) and its revised version, FGP1, should reflect the deterioration of a gearbox and track the gear tooth health condition over time. FGP is defined as the part (percentage of points) of the residual error signal which exceeds three standard deviations calculated from the baseline residual error signal taken when the run began, or

$$FGP = 100 \sum_{i=1}^{L} \frac{1}{L} I(r_i > \overline{r} + 3\sigma_0), \tag{2.2}$$

where r_i's are the residual error signal points. A series of Morlet wavelets are used as a filter to decompose the original vibration signal and obtain the harmonic error signal composed of vibration components from both the pinion and the gear.The residual error signal is the sum of vibration components that are purely due to the pinion and vibration components that are purely due to the gear [54]. $\overline{r}$ is the mean value of the current residual signal, σ_0 is the standard deviation and $I(\cdot)$ is the indicator function defined as

$$I(x > x_0) = \begin{cases} 1 & if\ x > x_0, \\ 0 & if\ x \leq x_0. \end{cases} \tag{2.3}$$

The current residual signal is compared with its mean value to adjust for possible changes in the running conditions, such as change in load, assuming that this should not affect the standard deviation. FGP1 is defined as the weighted part (weighted percentage of points) of the residual error signal, which exceeds three standard deviations from the baseline residual error signal, or

$$FGP = 100 \sum_{i=1}^{L} \frac{w_i}{W} I(r_i > \overline{r} + 3\sigma_0), \tag{2.4}$$

where $w_i = I(r_i \leq \overline{r} + 3\sigma_0) + (\lfloor \frac{r_i - \overline{r}}{3\sigma_0} - 1 \rfloor + 1)$, $W = \sum_{i=1}^{L} w_i$, and $\lfloor \cdot \rfloor$ is the floor

function. For a normally distributed signal, 99.7% of the signal should remain within three standard deviations. When a gearbox is in good condition, the vibration signal is random; thus it should conform to a normal distribution and only 0.3% of its points have the probability of exceeding three standard deviations. At this time, the value of FGP and FGP1 should be very low. With the development of gear tooth faults, more and more abnormal points will appear making both FGP and FGP1 increase significantly.

Other time-domain processing techniques include time synchronous average (TSA) [16, 39], the autoregressive (AR) model, the autoregressive moving average (ARMA) model [4, 10], and principal component analysis (PCA) [7].

2.1.2 Frequency-domain Analysis

Frequency-domain analysis is based on the transformed signal in the frequency domain. The advantage of frequency-domain analysis over time-domain analysis is its ability to easily identify and isolate certain frequency components that are of interest. The most widely used conventional analysis is the spectral analysis by means of fast Fourier transform (FFT). Fourier proved that any periodic function $x(t)$ can be represented as a sum of sinusoids with frequencies which are integer multiples of the frequency of $x(t)$, i.e. if a continuous-time function, $x(t)$, is periodic with a time period of T_0, it can be represented by a Fourier series as in [9]

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{j\Omega_0 kt}, \tag{2.5}$$

where Ω_0 is the frequency of the function given by $\frac{2\pi}{T_0}$, and c_k's are coefficients given by

$$c_k = \frac{1}{T_0}\int_{t_0}^{t_0+T_0} x(t)e^{-j\Omega_0 kt}dt. \tag{2.6}$$

For non-periodic signals there is also a representation in terms of complex exponentials which is called a continuous-time Fourier transform (CTFT) [42]:

$$X(\Omega) = \int_{-\infty}^{\infty} x(t)e^{-j\Omega t}dt. \tag{2.7}$$

These concepts can be extended to discrete-time signals by using discrete-time Fourier transform (DTFT); that is,

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}dt, \tag{2.8}$$

where n is the length of the discrete series. The discrete Fourier transform is a sampled version of DTFT that can be processed by computers. An N-point DFT is defined as,

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}kn}, 0 \leq k \leq N-1. \tag{2.9}$$

Fourier spectral analysis has provided a general method for examining global energy-frequency distributions. As a result, the term "spectrum" has become almost synonymous with the Fourier transform of data because the Fourier transform has dominated data analysis efforts since its introduction, and has been applied to almost all kinds of data [2, 44, 53]. The most commonly used tool in spectral analysis is the power spectrum. It is defined as $S_{xx}(k) = X(k)X^*(k)$, where "$*$" denotes the complex conjugate. The complex conjugate of a complex number, $z \equiv a + bi$, is $\overline{z} \equiv a - bi$. Fourier transform is commonly used as the first step in signal processing because it provides a basic sense of the collected signals. It can, however, reflect only averaged spectra in the frequency domain and it gives no information regarding time for some

observed peaks.

Besides the wide acceptance of the power spectrum, other useful spectra for signal processing have been developed and have been shown to have their own advantages over the FFT spectrum in certain cases. Cepstrum has the ability to detect harmonics and sideband patterns in a power spectrum. There are several versions of cepstrum [24], among them, a power cepstrum, which is defined as the inverse Fourier transform of the logarithmic power spectrum, or

$$\widetilde{s} = FT^{-1}[ln(S_{xx}(k)]. \tag{2.10}$$

The cepstrum is highly sensitive to recurrent patterns such as those generated by local faults. It can be useful in interpreting the spectrum and as a tool for the detection of periodic structures [3]. It has been found, however, that cepstrum is not sensitive to the different progression levels of same faults [16].

Other frequency-domain methods include high order spectrum [52], AR spectrum [17], and ARMA spectrum [47] based on the AR model and the ARMA model respectively.

2.1.3 Time-frequency Domain Analysis

One limitation of frequency-domain analysis is that it is unable to handle non-stationary waveform signals, which are very common when faults occur on machines [29]. Thus, time-frequency analysis, which investigates waveform signals in both the time and frequency domains, has been developed for non-stationary waveform signals.

The simplest way of performing a time-frequency analysis is certainly to consider a non-stationary signal as a series of quasi-stationary segments for which the stationary assumption is justified in each segment. Short-time

Fourier transform (STFT, also called spectrogram) offers a constant resolution in the time as well as in the frequency domain [13]. The transform is expressed as

$$X_F(b,\omega) = \int_{-\infty}^{+\infty} w^*(t-b)x(t)e^{j\omega t}dt, \tag{2.11}$$

where $w(\cdot)$ is the window function, and b and ω are the time and frequency scale respectively. This method restricts the Fourier transform within a specified window which slides along the time axis. Wang and McFadden [51] applied the STFT to the calculation of the time-frequency distribution of a gear's vibration signals. They showed that selecting the Gaussian function as the window function is suitable for the calculation of the spectrogram, giving a representation which is free from ripple and easy to interpret. It is a drawback mentioned in [13], however, that a good frequency resolution can be achieved only by means of a large window, which results in poor time resolution; conversely, good time resolution implies a short window, which results in poor frequency resolution.

Wavelet theory has developed rapidly in many areas in the past decade due to its flexibility and its efficient computational implementation. The continuous wavelet transform (CWT) of a square-integrable (i.e. $\int_{-\infty}^{\infty}|x(t)|^2dt$ being finite) and continuous-time signal $x(t)$ is the inner product between $x(t)$ and the wavelet, $\Psi_{a,b}(t)$, which gives wavelet coefficients [59]

$$X_w(a,b) = \frac{1}{\sqrt{|a|}}\int_{-\infty}^{+\infty} x(t)\Psi^*(\frac{t-b}{a})dt, \tag{2.12}$$

where $\Psi^*(t)$ is the complex conjugate of the mother wavelet function, $\Psi(t)$; a is the scale parameter; and b is the time parameter. Wavelet transform investigates the similarity of the original signals to a set of scaled and shifted versions

of the mother wavelet. Types of mother wavelets include Haar, Daubechies, Symlets, Coiflets, Meyer, Mexican Hat, and Morlet wavelet. They have been widely applied in the detection of gear faults [6, 36, 55]. Corresponding to the DTFT, there is also a discrete wavelet transform (DWT) but the discretized parameters are scale and time parameters, i.e. $a = 2^j$, $\frac{b}{2^j} = k$, and

$$X_w(k,j) = \frac{1}{\sqrt{2^j}} \int_{-\infty}^{+\infty} x(t)\Psi^*(\frac{t-2^j k}{2^j})dt, \tag{2.13}$$

where j and k are integers. The DWT behaves like a filter bank that decomposes an original signal, $x(t)$, into Approximation Coefficients, A_1, and Detail Coefficients, D_1, at the first level by a low pass filter and a high pass filter respectively. A higher level approximation vector, A_{j-1}, is decomposed into A_j and D_j again until the level reaches a preset number, J. The wavelet tree for $J = 3$ is illustrated in Fig. 2.1 [41]. The application of the DWT in fault

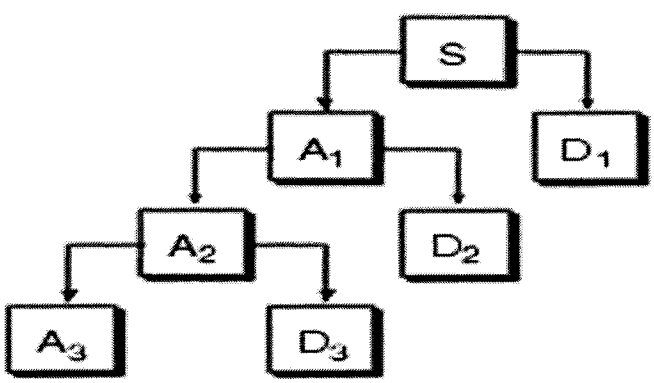

Figure 2.1: A three-level wavelet tree [41].

detection of spalling in bearings, worn gears and washing machines can be found in [22, 40, 49].

A major problem of wavelet is its non-adaptive nature. Once the mother

wavelet is selected, one has to use it to analyze all the data. It can be said that the selection of basis of the decomposition is *a priori*. In reality, we cannot have enough data to cover all possible points in the phase plane, therefore, most of the signals to be dealt with are transient in nature so that non-stationary and non-linear properties cannot be ignored. Locality and adaptivity are the necessary conditions on which to base an expanding non-linear and non-stationary time series. To deal with this problem, a new manner is introduced in [27] where a signal is written as a finite sum of intrinsic mode functions (IMFs). The method of obtaining this decomposition is called the empirical mode decomposition (EMD) method. The method is adaptive and is driven by the signal itself. Details of the EMD method will be given in Section 2.2.

Other time-frequency domain techniques include Choi-William distribution [14], Wigner-Ville distribution (WVD) [15], and S-transform [48].

2.2 The Empirical Mode Decomposition Method

In the last section, we reviewed some time-frequency domain signal processing methods. In contrast to all the previous methods, Huang *et al* [27] introduced a new method that is intuitive, direct, *a posteriori* and adaptive, with the basis of the decomposition based on, and derived from, the data. Actually, Huang *et al* proposed a general approach which requires two steps in analyzing the data. The first step is the empirical mode decomposition (EMD) which decomposes the data into a number of intrinsic mode function (IMF) components, thus expanding the data on a basis derived from itself. The second step is the Hilbert spectral analysis (HSA) which applies the Hilbert transform to the decomposed IMFs and constructs an energy-frequency-time distribution, designated as the Hilbert spectrum, from which the time localities of events will

be preserved. In other words, this method uses the instantaneous frequency and energy rather than the global frequency and energy defined by Fourier spectral analysis. EMD is actually a pre-step of HSA that decomposes the data into components for which the instantaneous frequency can be defined.

2.2.1 Intrinsic Mode Functions

Physically, the necessary conditions for defining a meaningful instantaneous frequency are that the functions are symmetric with respect to the local zero mean, and have the same number of zero crossings and extrema. As a result, an intrinsic mode function (IMF) is defined by two conditions:

1. in the whole data set, the number of extrema and the number of zero crossings must either be equal or differ at most by one;

2. at any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero.

The first condition is obvious; it is similar to the traditional narrow band requirements for a stationary Gaussian process. The second requirement modifies the classical global requirement to a local one; it is necessary so that the instantaneous frequency will not have the unwanted fluctuations induced by asymmetric wave forms. A local mean is the curve that makes the data purely symmetric after being subtracted from the data. Ideally, the requirement should be "the local mean of the data being zero". For non-stationary data, the "local mean" involves a "local time scale" to compute the mean, which is impossible to define. As a substitution, "envelope mean" which is defined by the local maxima and the local minima is used to force local symmetry instead. This is a necessary approximation to avoid the definition of a local

averaging time scale. This type of approximation is called an "envelope-mean" approximation.

The name "intrinsic mode function" has been adopted because it represents the oscillation mode imbedded in the data. With this definition, the IMF in each cycle, defined by the zero crossings, involves only one mode of oscillation; no complex riding waves are allowed. With this definition, an IMF is not restricted to a narrow band signal, and it can be both amplitude and frequency modulated. In fact, it can be non-stationary. As discussed above, purely frequency or purely amplitude modulated functions can be IMFs even though they have a finite bandwidth according to the traditional definition. A typical IMF is shown in Fig. 2.2 (Fig. 2 of [27]).

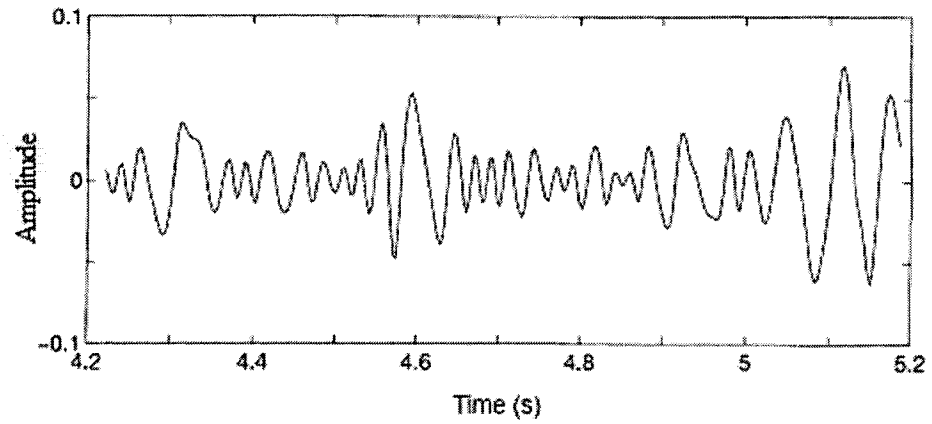

Figure 2.2: A typical intrinsic mode function with the same number of zero crossings and extrema, and symmetry of the upper and lower envelopes with respect to zero [27].

2.2.2 The Sifting Process

Unfortunately, most of the data are not IMFs. At any given time, the data may involve more than one oscillatory mode so that we have to decompose the

data into IMF components. The decomposition is based on three assumptions:

- the signal has at least two extrema–one maximum and one minimum;
- the characteristic time scale is defined by the time lapse between the extrema;
- if the data is totally devoid of extrema but contains only inflection points, then it can be differentiated one or more times to reveal the extrema. Final results can be obtained by integration(s) of the components. Actually, for vibration and acoustic signals, this case is rarely seen.

The time lapse between successive extrema is adopted as the definition of the time scale for the intrinsic oscillatory mode, not only because it gives a much finer resolution of the oscillatory modes, but also because it can be applied to data with a non-zero mean, either all positive or all negative values, without any zero crossings. A systematic way of extracting the oscillatory modes, called the sifting process, is described as follows.

By virtue of the IMF definition, the decomposition method can simply use the envelopes defined by the local maxima and minima separately. Once the extrema are identified, all the local maxima, $E_{max}(t)$, are connected by a cubic spline line as the upper envelope, and all the local minima, $E_{min}(t)$, are connected by a cubic spline line as the lower envelope as well. Their mean is denoted as m_{11}, and the difference between the data and m_{11} is the first component, h_{11}, i.e.

$$x(t) - m_{11} = h_{11}. \tag{2.14}$$

Here, the symbols m_{jk} and h_{jk} mean they are variables obtained for the jth decomposition level and kth iteration operation. The first sifting process is

illustrated in Fig. 2.3 (a)-(c) (Fig. 2.3 (a) gives the original signal; Fig. 2.3 (b) gives the data in the thin solid line, the upper and the lower envelopes in the dot-dashed lines, and their mean in the thick solid line, which bisects the data; and Fig. 2.3 (c) gives the difference between the data and the local mean as in equation (2.14)) .

Ideally, h_{11} should be an IMF, because the construction of h_{11} described above seems to have been made to satisfy all the requirements of IMF. In reality, however, the cubic spline interpolation can generate new extrema, and shift or exaggerate existing ones. The sifting process serves two purposes: to eliminate riding waves, and to make the wave profiles more symmetric. Toward this end, the sifting process has to be repeated a number of times. In the second sifting process, h_{11} is treated as a new signal, then

$$h_{11} - m_{12} = h_{12}, \tag{2.15}$$

where m_{12} is the mean of the upper and lower envelopes of h_{11}. We can repeat this sifting procedure k times, until h_{1k} is an IMF, that is

$$h_{1(k-1)} - m_{1k} = h_{1k}, \tag{2.16}$$

and the result, the first IMF from the signal, is denoted as

$$c_1 = h_{1k}. \tag{2.17}$$

We should have a criterion to determine when to stop the process of finding the first IMF and subsequent IMFs. It should not be too extreme because IMF components should still retain enough physical sense of both amplitude

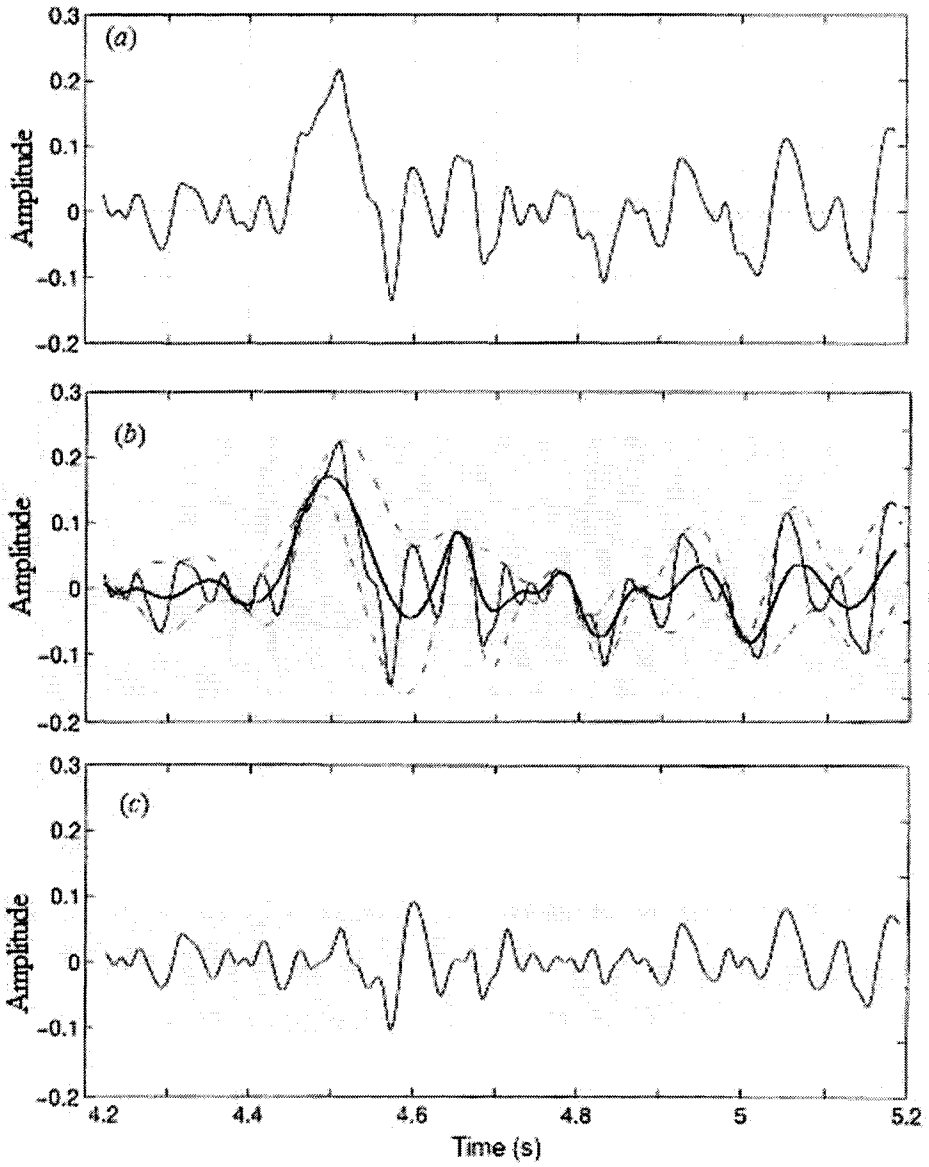

Figure 2.3: Illustration of the sifting process (Fig. 3 of [27]).

and frequency modulations. This can be accomplished by limiting the size of the standard deviation, SD, computed from two consecutive sifting results as

$$SD = \sum_{t=0}^{T} \left[\frac{|h_{1(k-1)}(t) - h_{1k}(t)|^2}{h_{1(k-1)}^2(t)} \right]. \tag{2.18}$$

A typical value for SD can be set between 0.2 and 0.3 [27].

Overall, c_1 should contain the finest scale or the shortest period component of the signal. We can separate c_1 from the rest of the data by

$$x(t) - c_1 = r_1. \tag{2.19}$$

Since the residue, r_1, still contains information of longer period components, it is treated as new original data and subjected to the same sifting process as described above. This procedure can be repeated on all the subsequent r_j's, and all the subsequent c_j's are obtained by using the same stopping criterion as the first IMF. The result is

$$\left.\begin{aligned} r_1 - c_2 &= r_2 \\ &\vdots \\ r_{n-1} - c_n &= r_n \end{aligned}\right\}. \tag{2.20}$$

The whole sifting process can be stopped by any of the following predetermined criteria: when the component, c_n, or the residue, r_n, becomes so small that it is less than the predetermined value of substantial significance, or when the residue, r_n, becomes a monotonic function from which no more IMFs can be extracted. To distinguish this stopping criterion from the one that stops iterations of finding each IMF, we refer the "stopping criterion/criteria" to only the one that stops iterations of finding each IMF (unless specified otherwise).

Even for data with zero mean, the final residue can still be different from zero; for data with a trend, the final residue should be that trend. By summing up equations (2.19) and (2.20), we finally obtain

$$x(t) = \sum_{i=1}^{n} c_i + r_n. \tag{2.21}$$

Thus, we have achieved a decomposition of the data into n empirical modes, and a residue which can be either the mean trend or a constant.

A set of vibration signals from a gearbox experiment [50] is used to exemplify the sifting process. The experiment system is shown in Fig. 2.4. The sampling frequency is 2560Hz. The length of the data is 3.2 seconds. The rotation speed of the motor is 600RPM. A damaged gear (Gear 4) seeded with a fault, a missing tooth, meshes with a normal gear (Gear 3) . The original collected data is shown in Fig. 2.5 and characteristic frequencies of the system are shown in Table 2.1. The decomposition result is shown in Fig. 2.6 which is a decomposition.

Table 2.1: Characteristic frequencies of the gearbox [50].

Input shaft	Middle shaft	Output shaft	Gears 1&2 meshing	Gears 3&4 meshing
10Hz	3.3Hz	5.5Hz	160Hz	133Hz

2.2.3 Critical Issues in the EMD

In the previous section, we described the most important part of the sifting process: the approximation to the local mean of a signal by upper and lower envelopes obtained from a cubic spline interpolation. A good approximation can capture the general trend of the signal while keeping the local features as much as possible. On the contrary, a poor approximation will lose much infor-

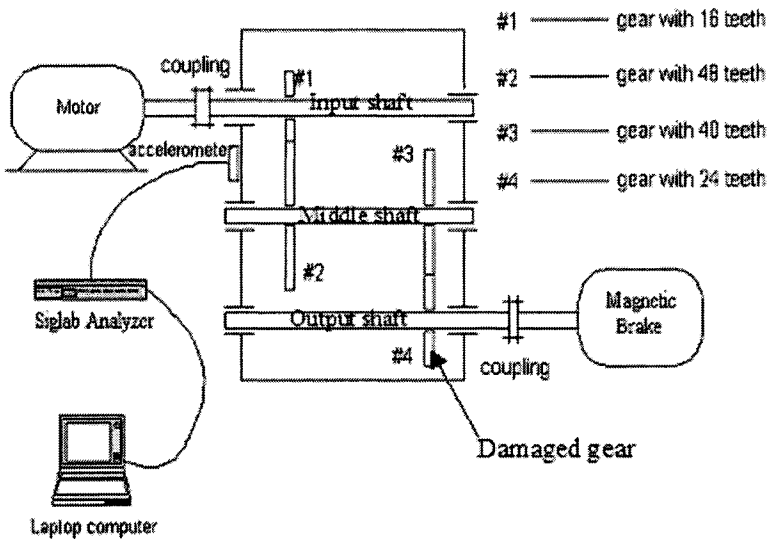

Figure 2.4: Illustration of an experimental system [50].

mation within the original signal causing the decomposed IMFs to have less physical meaning. Thus, approximation to the local mean is a critical issue in EMD, one for which there is considerable room for improvement. In Chapter 3 and Chapter 4 we will improve upon this critical issue using two types of approximation. In Chapter 3, the monotone piecewise Hermite interpolation will be applied to the envelope approximation procedure to preserve the monotonicity of the data [19]. As mentioned in [27], the envelope-mean interpolation creates a problem called end point swings. This problem exists because the length of the given data is finite and the end points cannot be maximum and minimum points simultaneously. Since the monotone piecewise Hermite interpolation also serves the envelope-mean approximation, the problem still exists. Approaches dealing with this problem will be discussed at length. In Chapter 4, a direct approximation approach using windowed local mean method will be applied without constructing envelope means; this method performs better

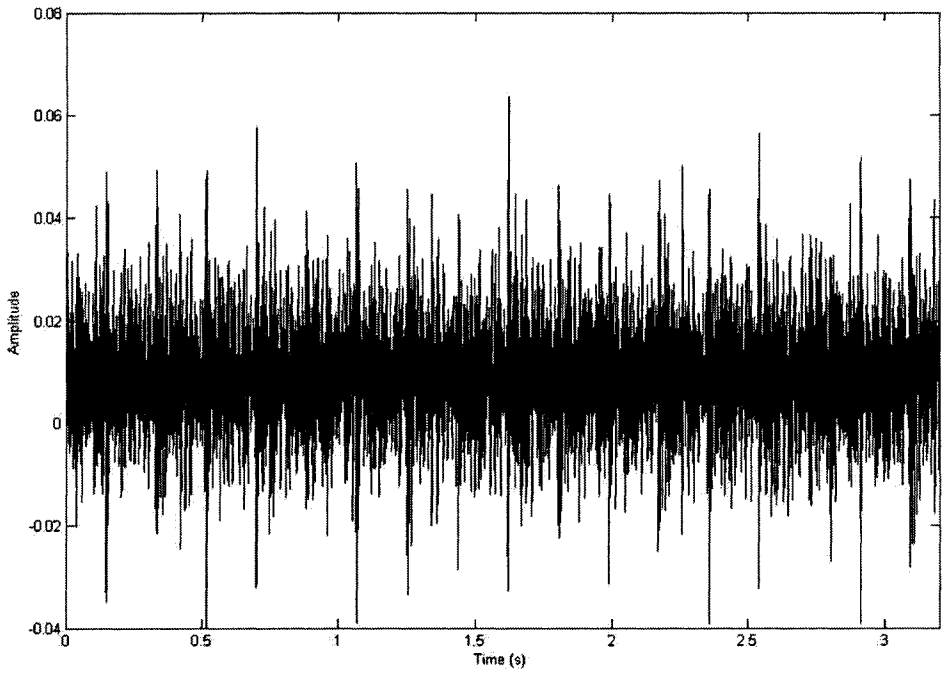

Figure 2.5: Waveforms of signals collected from a gearbox with tooth missing fault.

when identifying impulses [45]. The problem of end point swings does not exist here but a corresponding end point extension procedure for this method will be considered.

2.2.4 Analysis of Obtained IMFs

We have discussed the basic idea of EMD and some of its critical issues. Although this thesis concentrates on improving EMD, clearly the decomposition procedure is not the final step of signal processing. The purpose of conducting the decomposition is to utilize its result for fault detection or other applications so analysis of obtained IMFs is necessary. Visual observation is the simplest and most direct way of doing this. In [27] Huang *et al* introduced an

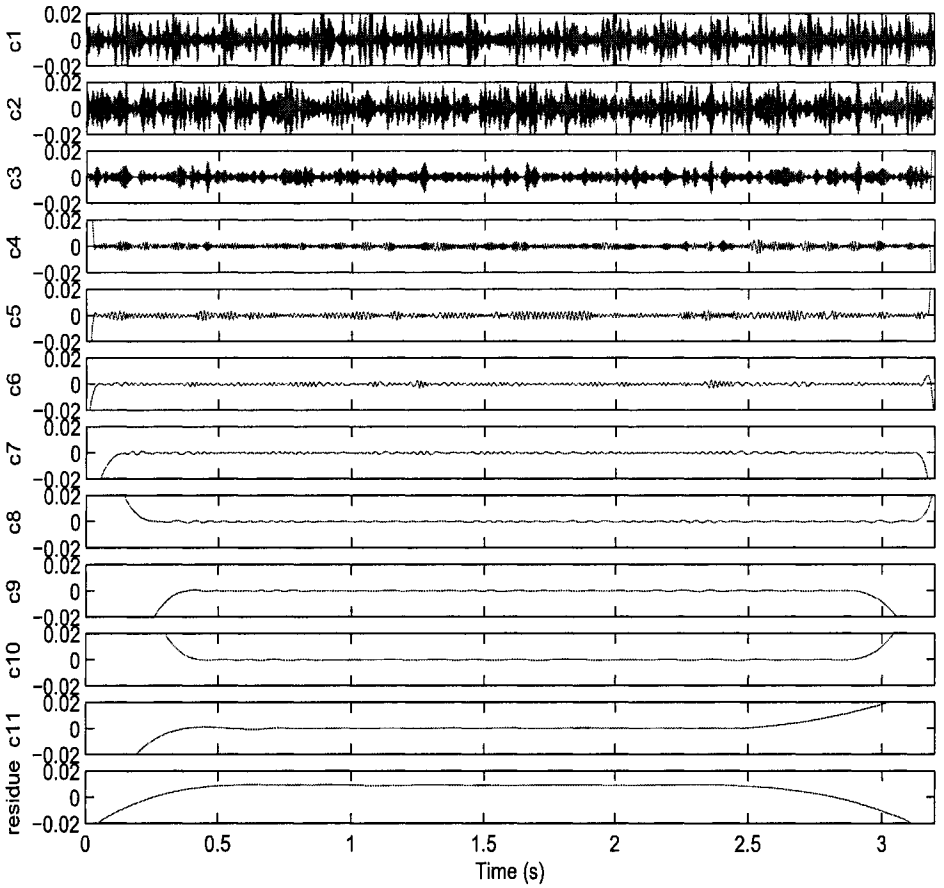

Figure 2.6: Illustration of the decomposition of a vibration data set after the sifting process.

advanced analysis to IMFs that conducts the Hilbert spectral analysis using Hilbert transform (HT) as the second step of their integrated approach. Since this is not the focus of this thesis, we provide just a brief review of HT and instantaneous frequency.

The Hilbert transform was first developed to process non-stationary narrow-band signals [23]. The Hilbert transform is a time-series analysis technique for deriving amplitude and phase information from a data set as a function of

time. It is a powerful tool for dealing with non-stationary signals. For an arbitrary time series, $x(t)$, we define its Hilbert transform, $y(t)$, as [45]

$$y(t) = H[x(t)] = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{x(t')}{t-t'}dt'. \tag{2.22}$$

According to the definition of Hilbert transform, $y(t)$ is just a time-delay and 90° phase-shifting of the original signal. It means that $y(t)$ is delayed by a quarter of a period in time domain than the signal $x(t)$. With their definitions, $x(t)$ and $y(t)$ form a complex conjugate, $Z(t)$, as

$$z(t) = x(t) + y(t) = a(t)e^{i\theta(t)}, \tag{2.23}$$

where $a(t) = \sqrt{x^2(t) + y^2(t)}$ and $\theta(t) = arctan(\frac{y(t)}{x(t)})$. Based on the expression of the Hilbert transform, Huang *et al* proposed a definition of the instantaneous frequency for narrow band signals as

$$\omega = \frac{d\theta(t)}{dt}. \tag{2.24}$$

Then, with the definition, instantaneous frequencies are calculated for each IMF obtained from the original signal. Let c_1, c_2, $\cdots$, c_n be n IMFs of $x(t)$ generated by the EMD method. Calculating the Hilbert transform of each mode gives n complex functions [45]:

$$\begin{aligned} z_1(t) &= c_1(t) + iH[c_1(t)] = a_1(t)e^{i\theta_1(t)} \\ &\vdots \\ z_n(t) &= c_n(t) + iH[c_n(t)] = a_n(t)e^{i\theta_n(t)}. \end{aligned} \tag{2.25}$$

The frequency of each mode is described by $\theta_j'(t)$. So, all $\theta_j'(t)$, $j = 1, \cdots, n$

together give a time-frequency analysis of the signal $x(t)$, which is called the Hilbert spectrum. As an example, the Hilbert spectrum of the first IMF of the signal used in Fig. 2.6 is shown in Fig. 2.7.

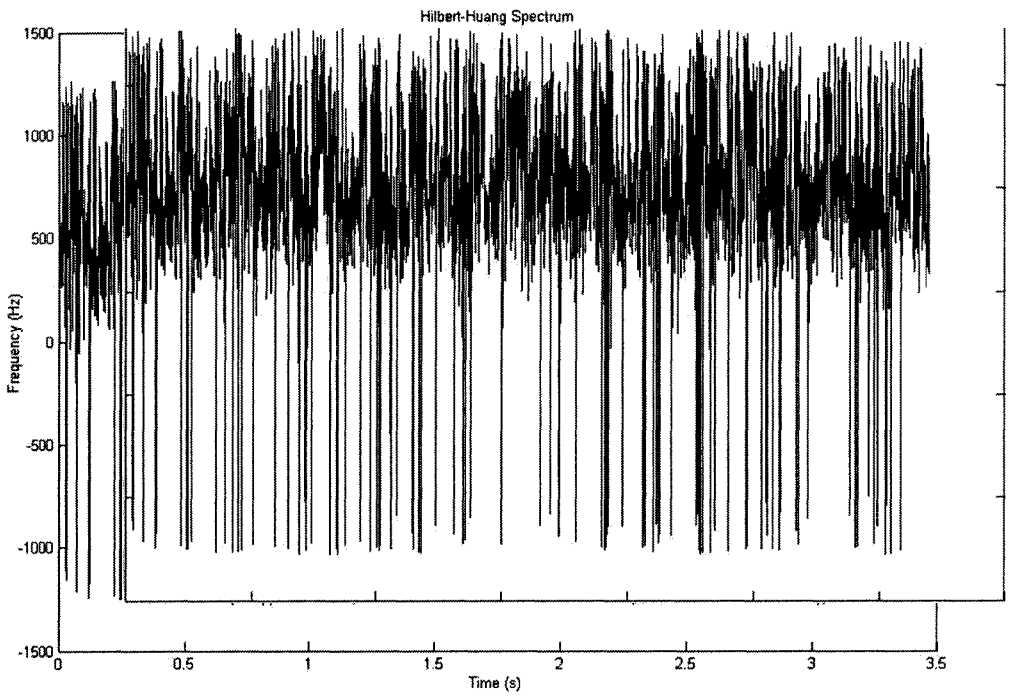

Figure 2.7: The Hilbert transform of the first IMF of the signal given in Fig. 2.6.

Zuo and Fan [63] also proposed an IMF analysis method that combines EMD with Hilbert transform. Unlike the Hilbert-Huang transform, this method takes the change of amplitude of the IMF's envelope spectrum into account other than the instantaneous frequency. It is compared with both the Hilbert-Huang transform and the wavelet transform using simulated signals and real signals collected from a gearbox. The results obtained show that the proposed method is effective in capturing hidden fault impulses.

2.2.5 Applications of the EMD

Applications for EMD have been found in many areas. In astrophysics, Komm *et al* [32] applies EMD and Hilbert analysis to time series of rotation residuals at all latitudes and at all depths in the solar convection zone derived from 49 Global Oscillation Network Group data sets covering the period May 7 1995 to May 15 2000. They calculate Hilbert power spectra for each time series in order to determine whether the rotation rate in the convection zone shows any other systematic temporal variation besides the so-called torsional oscillation pattern in the upper convection zone and the periodicity of 1.3 years near the base of the convection zone. In mechanical systems, a method is presented in [37] for monitoring the evolution of gear faults based on EMD. Experimental vibration signals from a test rig have been decomposed into IMFs. An empirical law, which relates the energy content of the intrinsic modes to crack magnitude, has been established. The modal energy is thus associated with the deterioration in gear condition and can be utilized for system failure prediction. Yu [60] used EMD to decompose the vibration signal of a roller bearing and establish the AR model of each IMF component. Practice examples show that the proposed approach can be applied effectively. In earthquake research, Zhang *et al* [61] uses EMD to analyze recordings of hypothetical and real wave motion, the results of which are compared with the results obtained by the Fourier data processing technique. The analysis of the two recordings indicates that the method is able to extract some motion characteristics useful in studies of seismology and engineering, which might not be exposed effectively and efficiently by the Fourier data processing technique.

Chapter 3

Improvement of the Envelope-Mean Approximation by Using Monotone Piecewise Hermite Interpolation

We have reviewed the concepts and basic algorithm of EMD in Chapter 2. The essential part of EMD is the sifting process using the envelope-mean approximation. The envelope-mean approximation includes these steps:

1. Identification of local maxima of the data being analyzed. For example, when checking three consecutive points, $P1$, $P2$ and $P3$, if $P2$ is greater than both $P1$ and $P3$, then $P2$ is identified as a maximum.

2. Connection of local maxima using an interpolation method to obtain an upper envelope. The cubic spline interpolation is used in [27].

3. Identification of local minima. The same approach is used as for maxima.

4. Connection of these local minima using a same interpolation method to obtain a lower envelope.

5. Use of the average of these two envelopes to approximate the local mean.

Although the effectiveness of this relatively new technique has been demonstrated in many applications, EMD itself is not perfect itself. Even the authors of the original paper have stated in [27], "At any rate, improving the spline fitting is absolutely necessary." Possible improvements of the envelope-mean approximation will be investigated in this chapter. Section 3.1 reviews reported interpolation methods with their merits and disadvantages, among them the monotone piecewise Hermite interpolation. In Section 3.2, we choose the monotone piecewise Hermite interpolation to replace the cubic spline interpolation in EMD; comparing their performance using simulated signals on a common standard. The performance of the winner of the comparison on an experimental data is tested in Section 3.3. A summary can be found at the end of the chapter.

3.1 Reported Interpolation Methods

Interpolation is the process of constructing a function that takes on given values at a given data set. Such a function is called the interpolating function or interpolant and can be seen as an approximation of a function which is known at some points [30]. These known or given points are also called knots. Let $\{x_i\}_{i=1}^n$ be a partition of the interval $I = [a, b] \subset \mathbf{R}$, i.e. $a = x_1 < \cdots < x_n = b$. Suppose that a function, g, is given in I, the problem we consider is finding a function, $f = f(x)$, such that

$$f(x_i) = g(x_i), i = 1, \cdots, n. \tag{3.1}$$

Many interpolation methods have been reported since [30] which was published in 1910. For historical as well as pragmatic reasons, the most important

class of interpolating functions is a set of polynomial functions. Polynomial functions have the advantage of being easy to evaluate directly and can be easily added, multiplied, integrated, or differentiated [30]. Polynomial functions have two branches: one is global polynomial interpolation which passes one polynomial function through all the data. If we want f to be a polynomial we call $f(x) = p(x)$ a global polynomial interpolating function. The linear space of polynomials of order n is denoted by

$$P_n = \{p | p(x) = a_0 + \cdots + a_n x^n, a_i \in \mathbf{R}\}. \tag{3.2}$$

A classical result from algebra is that there is a unique polynomial, p, of order n such that $p(x_i) = g(x_i)$ for $i = 1, \cdots, n+1$. Another branch is piecewise polynomial interpolation which uses a polynomial function for each consecutive pair of knots. Instead of assigning f to be a single polynomial, f can be a polynomial on each interval, $[x_i, x_{i+1}]$, i.e. $f|_{[x_i, x_i+1]} = p_i$ with $p_{i-1}(x_i) = p(x_i)$.

A piecewise cubic interpolation is a piecewise interpolation that uses third order polynomial functions. Hermite cubic interpolation is a piecewise cubic interpolation that requires a continuous first derivative at knots [30]. If a continuous second derivative is also required, it is called a cubic spline interpolation. If a continuous second derivative is not required but some other conditions are supplied to guarantee monotonicity, the Hermite cubic interpolation is called a monotone piecewise Hermite interpolation [20]. Monotonicity is a basic property of a curve that reflects the increasing or decreasing feature of some consecutive points of the data. As an illustration, the two solid lines in Fig. 3.1 keep the monotonicity of points A and B but the dashed line doesn't,

since it generates a minimum between A and B. Fritsch and Butland [19] described a modified monotone piecewise Hermite interpolation that solves the three problems that were presented in the original interpolation approach [20]. Fig. 3.2 shows a clear understanding of the hierarchy of different interpolation approaches.

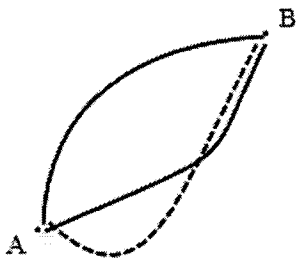

Figure 3.1: Illustration of monotonicity

Among those interpolation methods, the cubic spline and the high-order spline interpolation have been applied to EMD. Firstly, let's have a look at how the cubic spline interpolation works to fulfill the purpose of approximation.

3.1.1 Cubic Spline Interpolation

The term "spline" comes from a flexible strip (wooden or rubber) used by shipbuilders and draftsmen to draw smooth shapes [5]. In the mathematical field, if the polynomial is of order three, the interpolant, f, is called "piecewise cubic". The linear space of k times differentiable piecewise cubic functions on $I = [a, b]$, is expressed as

$$S_3^k([x]) = \{f | f|_{[x_i, x_i+1]} \in P_3 \; for \; i = 1, \cdots, n-1 \; and \; f \in C^k[a, b]\}. \quad (3.3)$$

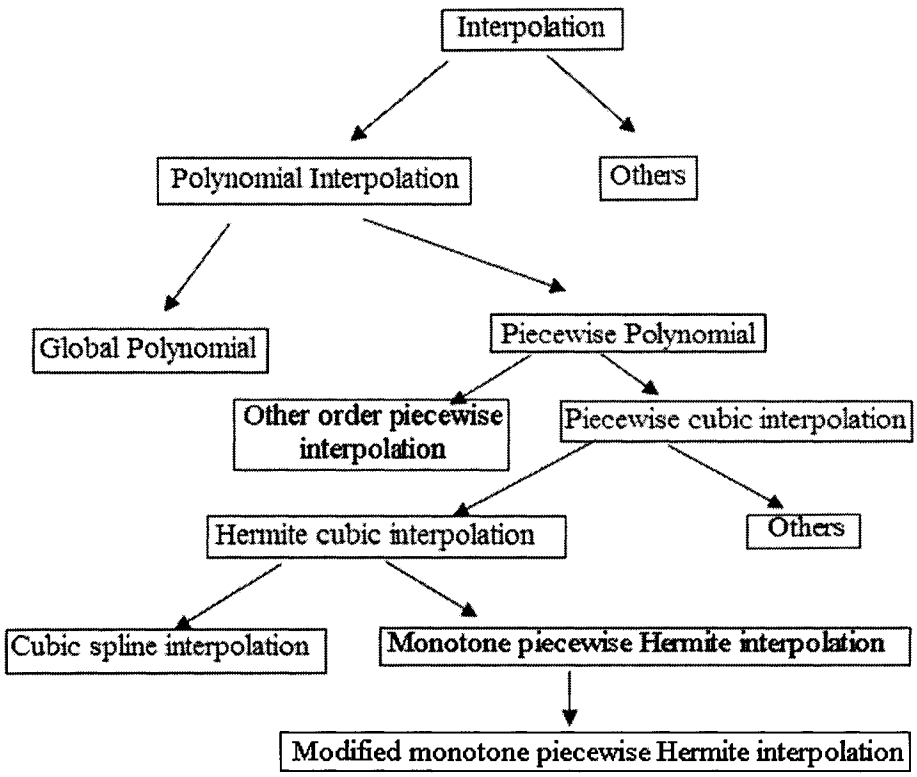

Figure 3.2: Classes of different interpolation approaches

In $S_3^k([x])$ and $k \leq 2$, we have infinite possible solutions for f such that $f(x_i) = g(x_i)$. For $k > 2$, there is no guarantee that such an f exists. For a solution of f, on each interval, $[x_i, x_{i+1}]$, f is given by $f|_{[x_i,x_{i+1}]} = p_i$, as

$$p_i(x) = c_{0,i} + c_{1,i}(x - x_i) + c_{2,i}(x - x_i)^2 + c_{3,i}(x - x_i)^3. \tag{3.4}$$

There are four coefficients, $c_{0,i}, c_{1,i}, c_{2,i}, c_{3,i}$, for each piecewise polynomial and there are $n - 1$ intervals. As a result, there are a total of $4(n - 1)$ unknown parameters. Remember there are two constraints on $p_i(x)$–$p_i(x_i) = g(x_i)$ and $p_i(x_{i+1}) = g(x_{i+1})$)–which contribute $2(n-1)$ constraints totally. Requiring f'

to be continuous adds $n-2$ constraints, i.e. one at each interior knot (a point not at ends). This means

$$f'(x_i) = p'_{i-1}(x_i) = p'_i(x_i). \tag{3.5}$$

We still need to have $4(n-1) - 2(n-1) - (n-2) = n$ constraints to obtain a unique solution for a piecewise cubic interpolating function. A special class of cubic interpolating functions are those which are twice differentiable, such functions are designated as cubic spline interpolants. This means that f satisfies

$$f''(x_i) = p''_{i-1}(x_i) = p''_i(x_i). \tag{3.6}$$

Thus, $n-2$ additional constraints have been added. The remaining two constraints give us a few types of cubic spline interpolation:

Complete cubic spline interpolant This specifies the derivative of $f(x)$ at the two end points, i.e. set $f'(x_1) = d_1$ and $f'(x_n) = d_n$ where d_1 and d_n are known values.

Natural spline interpolation This forces $f''(x_1) = 0 = f''(x_n)$. Physically this means that the graph of the spline is a straight line outside I.

'Not-a-knot' conditions These require $f'''(x)$ to be continuous at x_2 and at x_{n-2}, i.e. $f'''(x_2) = p'''_1(x_2) = p'''_2(x_2)$ and $f'''(x_{n-2}) = p'''_1(x_{n-2}) = p'''_2(x_{n-2})$.

Back to the application of piecewise cubic spline interpolation in EMD. It has the obvious merit of securing the continuity of the second derivative at knots. In [27], the 'Not-a-knot' type of interpolation was used to find an approximation to the local mean of a signal. For given collected data, however,

we usually don't know any features of its upper and lower envelopes if they are not revealed by the data itself. As a result, we prefer to have the interpolation represent the shapes of the envelopes as they are, i.e. to avoid the imposition of any additional details that are not confirmed by the data [12]. We can see from Figure 3.3 (picked out from Fig. 2.3), however, that both undershooting (shaded region A) and overshooting (shaded region B) problems occur due to the interpolation. In other words, interpolated splines of some consecutive monotonic maxima or minima do not maintain their monotonicity. One spline is not monotonic even between two consecutive maxima; there are bumps between the minima (in region A). The quality of the approximation of the local mean affects the iteration times of finding IMFs. A bad approximation will slow down the speed of decomposition. Also, the purpose of the decomposition is to identify IMFs before conducting Hilbert transform or other subsequent analysis. Low quality of the approximation will cause more serious asymmetry of IMFs that makes the results from consequent processes less meaningful.

3.1.2 High-order Spline Interpolation

Some research work has been done to find a better interpolation for doing the envelope-mean approximation. Huang *et al* suggested in [27] using high-order spline but the exploration was underway. Yang *et al* [58] proposed using the high-order spline interpolation to form the upper and lower envelopes. They examined quartic, quintic, six-order, and seven-order spline interpolations and found that the precision is improved in terms of IMF error (which was not defined). Interpolation becomes more and more time-consuming, however, as the order of polynomial increases. The undershooting and overshooting problem still exists since no conditions are added to take the monotonicity

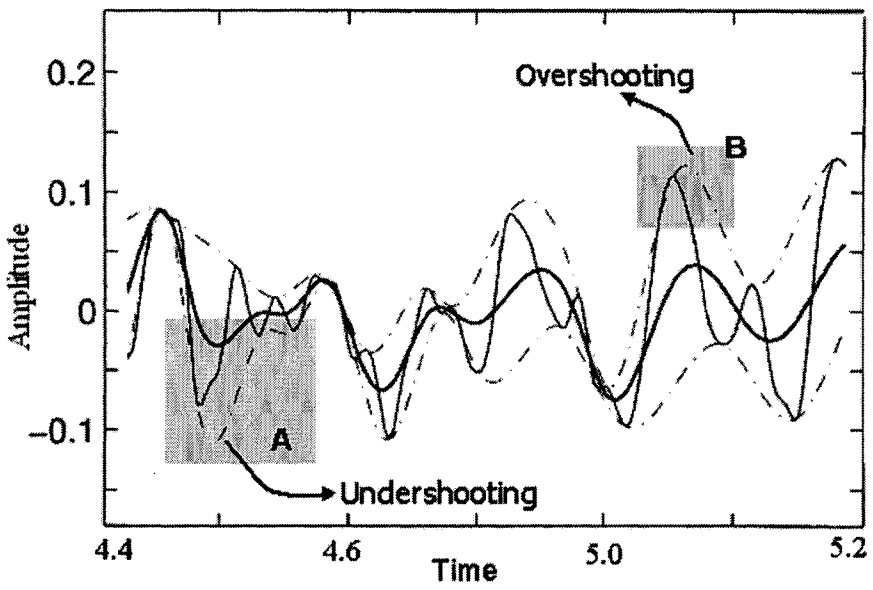

Figure 3.3: Example of the shortcomings of the cubic spline interpolation. The shaded regions indicate the segments on which monotonicity is not maintained.

into consideration.

3.1.3 Monotone Piecewise Hermite Interpolation

To have less of undershooting and overshooting mentioned above, it is preferable to keep some of the shape properties of a curve. In fact, there are a few shape properties that it may be advisable to preserve in order to have a good-looking interpolation. One of the most important ones is monotonicity [34]. The monotone piecewise Hermite interpolation is a modified version of the piecewise Hermite cubic interpolation that preserves the monotonicity of the data [19].

The piecewise Hermite cubic interpolation is a type of interpolation that

puts constraints on only the first derivative of the known knots, i.e.

$$f'(x_i) = p'_{i-1}(x_i) = p'_i(x_i) = d_i, \tag{3.7}$$

where d_i is the derivative value of the ith knot point. Hermite interpolation is a general base and is open to provision of various conditions that could give a unique solution, whereas cubic spline interpolation is just a special class of the piecewise Hermite cubic interpolation which implicitly constrains the value of d_i by conditions of second derivatives.

Some studies have been done to provide alternative conditions for the piecewise Hermite interpolation so that monotonicity of the data is kept. Monotonicity is a basic property of a curve; if it is not guaranteed an interpolant will deviate essentially from the curve. That is why monotonicity is the primary concern in this study. Let $a = x_1 < \cdots < x_n = b$ be a partition of the interval $I = [a, b]$. $I_i = [x_i, x_{i+1}]$ is a subinterval of I. Let $g_i : i = 1, 2, \cdots, n$ be a given set of data at knots. Our goal is to construct piecewise cubic functions, $p_i(x) \in C^1[I_i]$, on the partition of the interval such that if $f_i \leq f_{i+1}$ ($f_i \geq f_{i+1}$), $p_i(x)$ is an increasing (or decreasing) function.

Fritsch and Carlson [20] derived necessary and sufficient conditions for a cubic function to be monotonic in an interval. Let $\Delta_i = (g_{i+1} - g_i)/h_i$ be the slope of the line segment joining the data to be interpolated, where $h_i = x_{i+1} - x_i$. The piecewise function $p_i(x)$ can be expressed as

$$p_i(x) = \left[\frac{d_i + d_{i+1} - 2\Delta_i}{h_i^2}\right](x-x_i)^3 + \left[\frac{-2d_i - d_{i+1} + 3\Delta_i}{h_i^2}\right](x-x_i)^2 + d_i(x-x_i) + g_i. \tag{3.8}$$

It is clear that a necessary condition for monotonicity is that

$$sgn(d_i) = sgn(d_{i+1}) = sgn(\Delta_i), \tag{3.9}$$

where $sgn(x)$ represents the sign of x. Let $\alpha_i = d_i/\Delta_i$ and $\beta_i = d_{i+1}/\Delta_i$ be the respective ratios of the end-point derivatives to the slope of the straight line connecting the ith and the $i+1$ knot. It is proved by [20] that equation (3.9) is a sufficient and necessary condition for $p_i(x)$ being monotonic if $\alpha_i+\beta_i-2 \leq 0$. If $\alpha_i + \beta_i - 2 > 0$, $p_i(x)$ is monotonic, at least one of the following conditions is satisfied:

$$2\alpha_i + \beta_i - 3 \leq 0;$$

$$\alpha_i + 2\beta_i - 3 \leq 0;$$

$$\phi(\alpha_i, \beta_i) \geq 0,$$

where $\phi(\alpha, \beta) = \alpha - \frac{(2\alpha+\beta-3)^2}{3(\alpha+\beta-2)}$. As a consequence, a region, M, of acceptable values for α_i and β_i (hence d_i and d_{i+1}) is constructed to produce a monotonic interpolant on I_i. This region is shown in Fig. 3.4 (Fig. 1 of [20]). Using this result, a two-step procedure has been suggested for constructing such an interpolation. Unfortunately, however, the procedure suffers from three defects.

1. It requires two passes over the data, i.e. d_i should be initialized and then should be adjusted to d_i^* whenever $(\alpha_i, \beta_i) \notin M$ such that $(\alpha_i^*, \beta_i^*) \in M$.

2. The result is dependent on the order in which the data points are processed. The common order for processing data points is either from point 1 up to n or from n down to 1.

3. A correction introduced in the first interval could ripple through the entire interpolant.

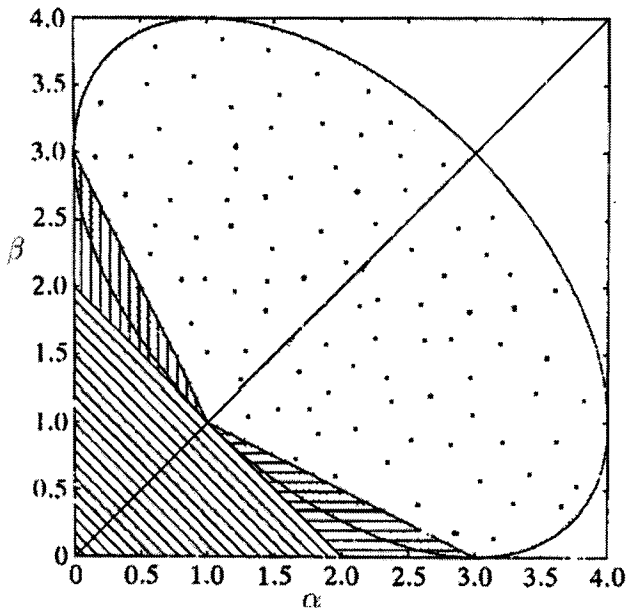

Figure 3.4: The monotonicity region, M, is a combination of these regions: diagonal hatching: $\alpha + \beta - 2 \leq 0$; vertical hatching: $\alpha + \beta - 2 > 0$ and $2\alpha + \beta - 3 \leq 0$; horizontal hatching: $\alpha + \beta - 2 > 0$ and $\alpha + 2\beta - 3 \leq 0$; dotted: $\phi(\alpha, \beta) \geq 0$; unshaded: cubic is non-monotone out of region M.

Fritsch and Butland [19] described a modified algorithm that avoids all of these problems. A function, G, is constructed such that

$$d_i = G(\Delta_{i-1}, \Delta_i), \quad i = 2, \cdots, n-1. \tag{3.10}$$

Since d_i depends only on neighboring slopes, this method based on equation (3.10) does not need two passes over the data, and a correction of a slope value would affect only two neighboring derivative values. If G is a symmetric function of its arguments, the result will also be independent of the direction

of processing, either from the starting point to the end point or from the end point to the starting point. Sufficient conditions for an acceptable G function are given in [12] and a general expression of G is given as

$$G(S_1, S_2) = \begin{cases} 0 & if\ S_1 S_2 \leq 0, \\ sgn(S_1)\frac{m|S_1||S_2|}{|S_1|+(m-1)|S_2|} & if\ |S_2| \leq |S_1|, \\ G(S_2, S_1) & otherwise. \end{cases} \tag{3.11}$$

When $m = 2$, G becomes the harmonic mean

$$G_H(S_1, S_2) = \begin{cases} \frac{2S_1 S_2}{S_1+S_2} & if\ S_1 S_2 > 0, \\ 0 & otherwise. \end{cases} \tag{3.12}$$

Equation (3.12) restricts the values $(\alpha_i, \beta_i) = (d_i/\Delta_i, d_{i+1}/\Delta_i)$ to the small square $[0, 2] \times [0, 2]$ of Fig. 3.4. Although it falls into the acceptable monotonicity region, Filling out the maximal symmetric acceptable region is desirable because if $S_1 S_2 > 0$ and $|S_1| \gg |S_2|$, $G(S_1, S_2) = mS_2$ according to equation (3.11), and a larger m will result in less difference between two consecutive slopes. Thus, m is set equal to 3 because the maximal symmetric acceptable region is $[0, 3] \times [0, 3]$, i.e.

$$G_3(S_1, S_2) = \begin{cases} 0 & if\ S_1 S_2 \leq 0, \\ sgn(S_1)\frac{3|S_1||S_2|}{|S_1|+2|S_2|} & if\ |S_2| \leq |S_1|, \\ G(S_2, S_1) & otherwise. \end{cases} \tag{3.13}$$

Equations (3.7), (3.10) and (3.13) give the algorithm of the modified monotone piecewise Hermite interpolation (MMPHI, in short). The key difference between this interpolation and the cubic spline interpolation is that this one re-

leases the constraints of the continuity of the second derivatives, while adding constraints related to monotonicity to have a unique piecewise cubic interpolant. The modified monotone piecewise Hermite interpolation has been used to estimate unsaturated soil hydraulic properties [8] and simulated animations of fluid dynamics equations [28] but it has not been applied to EMD. In Section 3.2 the application of this method to the envelope-mean approximation of EMD will be discussed.

3.2 Application of the MMPHI to the Envelope-Mean Approximation

3.2.1 Motivation

The purpose of the interpolation in EMD is to show what the envelopes of the data look like. The interpolant should show what is contained in the data and nothing more. If the configuration has any peculiarities, these should be drawn clearly; if an obvious feature is presented, the interpolant should represent it in a suitable way [12]. Since monotonicity is a key shape property of a curve, we want to keep monotonicity of the data in our study. The monotone piecewise Hermite interpolation is not the only method that considers monotonicity. Wolberg [57] described methods that minimize the second derivative discontinuity while the algorithm of [19] guarantees continuity of the first derivatives. But Wolberg's methods involve linear and quadratic programming which consume a great deal of time. The linear piecewise interpolation is the simplest way to keep monotonicity; but, with this approach, a smooth curve's upper and lower envelopes, and therefore its envelope mean, can become too sharp, losing its smoothness. Since the interpolation of EMD is based on extrema which do not confirm any information about the continuity of the second derivatives,

only continuity of the first derivatives is considered for our application. Thus, our motivation for applying MMPHI to the envelope-mean approximation is that it is able to maintain monotonicity of the data and guarantees continuity of the first derivatives without too much complexity.

3.2.2 The Proposed Approach and Its Expectation of Improvement

Our proposed approach in this thesis is to use the monotone piecewise Hermite interpolation developed by [19] to replace the cubic spline interpolation in [27] for the purpose of approximating upper and lower envelopes in the process of IMF decomposition.

By applying MMPHI to the envelope-mean approximation instead of the cubic spline and high-order spline interpolations, we expect to see a more accurate decomposition of an original signal. This is because MMPHI has less undershooting and overshooting problems so that the local mean being approximated is more accurate. The indication of accuracy will be defined in the comparisons in next subsections.

We also expect MMPHI to have an advantage over the cubic spline interpolation with regard to CPU time. We can verify theoretically that doing one MMPHI interpolation operation is faster than doing one cubic spline interpolation operation. Mathematically, the difference between the two methods is that they are using different ways to determine the values of d_i. Working one more step from equations (3.3)–(3.5), we have [45]:

$$h_i d_{i-1} + 2(h_{i-1} + h_i) d_i + h_{i-1} d_{i+1} = 3(h_i \Delta_{i-1} + h_{i-1} \Delta_i), \quad for \quad i = 2, \cdots, n-1. \tag{3.14}$$

With the "not-a-knot" conditions defined on page 33, two more equations can

be supplied to give a unique solution to the cubic spline interpolation, as

$$h_2 d_1 + (h_1 + h_2) d_2 = \frac{h_1 + 2(h_1 + h_2) h_2 \Delta_1 + (x_1)^2 \Delta_2}{h_1 + h_2}, \tag{3.15}$$

and

$$h_{n-2} d_n + (h_{n-1} + h_{n-2}) d_{n-1} = \frac{h_{n-1} + 2(h_{n-1} + h_{n-2}) h_{n-2} \Delta_{n-1} + (x_{n-1})^2 \Delta_{n-2}}{h_{n-1} + h_{n-2}}. \tag{3.16}$$

Thus, the solution is fully determined by solving the linear system

$$\begin{pmatrix} c_{11} & c_{12} & 0 & 0 & 0 & 0 \\ c_{21} & c_{22} & c_{23} & 0 & 0 & 0 \\ \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & c_{(n-1)(n-2)} & c_{(n-1)(n-1)} & c_{(n-1)(n)} \\ 0 & 0 & 0 & 0 & c_{(n)(n-1)} & c_{(n)(n)} \end{pmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{bmatrix} = \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_{n-1} \\ U_n \end{bmatrix}, \tag{3.17}$$

where, say, $c_{12} = h_1 + h_2$ and $c_{(n)(n-1)} = h_{n-1} + h_{n-2}$, and (according to equations (3.14)– (3.16)).

$U_1 = \frac{h_1 + 2(h_1 + h_2) h_2 \Delta_1 + (x_1)^2 \Delta_2}{h_1 + h_2}$ and

$U_n = \frac{h_{n-1} + 2(h_{n-1} + h_{n-2}) h_{n-2} \Delta_{n-1} + (x_{n-1})^2 \Delta_{n-2}}{h_{n-1} + h_{n-2}}$.

Solving equation (3.17) involves a tridiagonal matrix which is a square matrix with nonzero elements only on the diagonal and slots horizontally or vertically adjacent to the diagonal. We need $O(11n)$ arithmetic operations since inversion of this matrix needs $O(7n)$ operations [1] and multiplication of two matrices needs $O(4n)$ operations. In contrast, MMPHI is obtained without solving a linear system [30] so the computational complexity depends only on that of equation (3.13) which, as a consequence, requires only $O(3n)$ arithmetic

operations. Therefore, for an individual operation MMPHI is faster than the cubic spline interpolation. Obviously, a total CPU time of a type of EMD is also dependent on the number of iterations of the sifting process, number of IMFs decomposed and other system set-up expenses. With a more accurate envelope-mean approximation, we will have fewer iterations so total CPU time can be improved as well. The high-order interpolation is even slower than the cubic spline interpolation [58].

3.2.3 Test of the Proposed Approach

3.2.3.1 Common Standard for the Test

To test the proposed approach and compare its performance with reported methods, we should make sure that a common standard is set up for all participants. For an integrated EMD procedure, there are a few variables open to be selected, such as stopping criteria, approaches to dealing with end point swings, and the number of decomposition levels needed. In fact, it is rare to see more than five decomposed IMFs being used to provide information for the diagnosis of devices; this is because amplitudes of the IMFs beyond five are very small [63]. We set the maximal decomposition level to ten so that all useful information can be included. Chapter 2 has mentioned two other important aspects that need to be considered: stopping criteria and end point swings. There have been several reported studies on them. Therefore, before testing the proposed method, we will carefully select a stopping criterion and an approach to the problem of end point swings from reported studies. They will be used by all methods that will be compared.

Let's look at stopping criteria first. The envelope-mean method in EMD has a tendency to produce IMFs with uniform amplitude [27]. As a result, the

stopping criteria in the sifting process needs to be chosen properly. One should avoid a too stringent criterion that the physical meaning of IMFs would be lost; on the other hand, one should also avoid a too loose criterion that components deviating too much from IMFs would be decomposed. In Chapter 2 (page 20) we introduced the criterion of standard deviation. Huang *et al* described a second criterion in [26], where the sifting is stopped when the number of zero-crossing points and extrema is the same number for S successive sifting steps. Typically, a value of $3 \leq S \leq 5$ has proved successful as the default stopping criterion [26]. Rilling *et al* [46] proposed another criterion based on two thresholds θ_1 and θ_2, aimed at guaranteeing globally small fluctuations in the mean while taking into account locally large fluctuations. The mode amplitude is introduced as

$$a(t) \equiv \frac{1}{2}(e_{max}(t) - e_{min}(t)), \tag{3.18}$$

where $e_{max}(t)$ and $e_{min}(t)$ are the upper and lower envelopes respectively. Then, the evaluation function is defined as

$$\sigma(t) \equiv |\frac{m(t)}{a(t)}|, \tag{3.19}$$

where $m(t)$ is the mean of upper and lower envelopes. The criterion is that the sifting process is iterated until $\sigma(t) < \theta_1$ for some prescribed fraction $(1-\alpha)$ of the total duration, and $\sigma(t) < \theta_2$ for the remaining fraction. One can typically set $\alpha = 0.05$, $\theta_1 = 0.05$ and $\theta_2 = 10\theta_1$ [46]. The comparison given later will show why the stopping criterion from [46] will be used in this thesis.

The presence of swings of the interpolation at the end points (including both first and last points) of a set of a data is due to the finite length of the

data. End points are usually not the last extrema. Even if they are, they cannot be a maxima and a minima simultaneously. So, at most one envelope can be determined based on the data at the end. Other values for the points beyond the first or the last extrema have to be obtained by extrapolation, but any extrapolation would lack physical basis because no information of extrema beyond the first and the last ones can be referred to. Fig. 3.5 is an example that shows the problem of end point swings. The segment between two vertical dotted lines is the data for analysis. The four arrows indicate the maxima and minima beyond the studied segment. Ideally, the envelopes (dash lines) would still head for these extrema beyond the segment, but without any information for interpolation, the envelopes are extrapolated much differently than what we would like. This phenomenon is called end point swings.

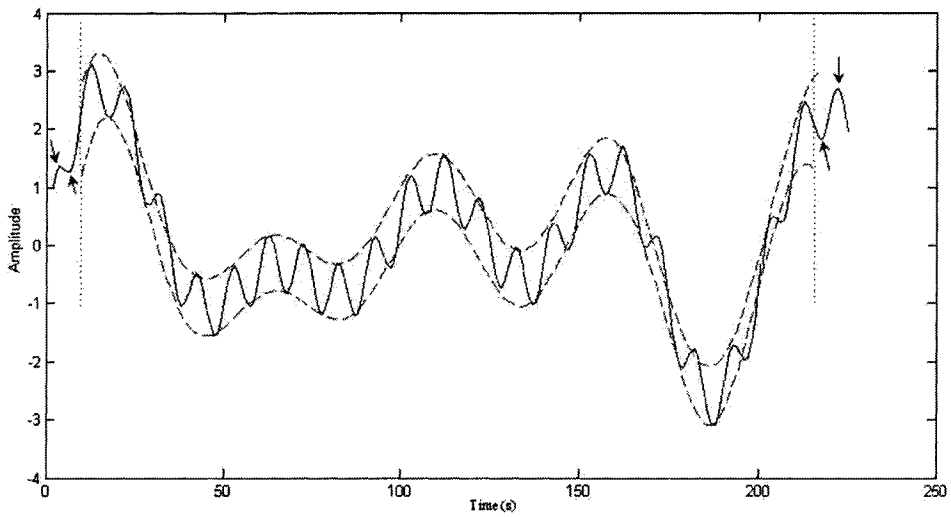

Figure 3.5: Illustration of end point swings. The dash lines are the upper and lower envelopes of the data.

We should not ignore this problem, otherwise, the large deviation could corrupt the data and propagate to the interior. A method of adding character-

istic waves at the end points which are defined by the two consecutive extrema for both their frequency and amplitude of the added waves is suggested in [27] but it does not specify the type of characteristic waves. Huang *et al* [25] specified extended waves as the added waves, which the first and the last two consecutive extrema are repeated beyond the first and the last data points at periods equal to the periods of the first and the last two maxima (if a maximum is the end extremum) or minima (if a minimum is the end extremum). Another approach is introduced both by Huang *et al* [25] and Zhao [62]. It uses the end extrema as two mirrors and reflects the data between the mirrors out of the end, doubling the length of the original data, which, as a result, doubles the time of the sifting process. Rilling's method [46] just mirrors the two end extrema of the data which is less time-consuming. This method will be proven best through the following comparison.

We can use a simulated signal to compare options for stopping criteria and end point swings and to pick suitable ones. The simulated signal is a combination of multiple sinusoid waves with different frequencies and amplitudes plus a global trend:

$$\begin{aligned} y(t) = \; & 0.5sin(2\pi * 1000t/10) + 2sin(2\pi * 1000t/50) + \\ & sin(2\pi * 1000t/120) + sin(2\pi * 1000t/200) + \\ & sin(2\pi * 1000t/300) + sin(2\pi * 1000t/500) + 0.0005t. \end{aligned} \tag{3.20}$$

The signal is shown in Fig. 3.6. We tested both a short series (3000 points) and a long series (8000 points) and set the time interval of any two adjacent points at 0.001s.

Two performance indicators are applied: one is the CPU time of EMD procedure, the other is the accuracy. The latter is represented by the mean

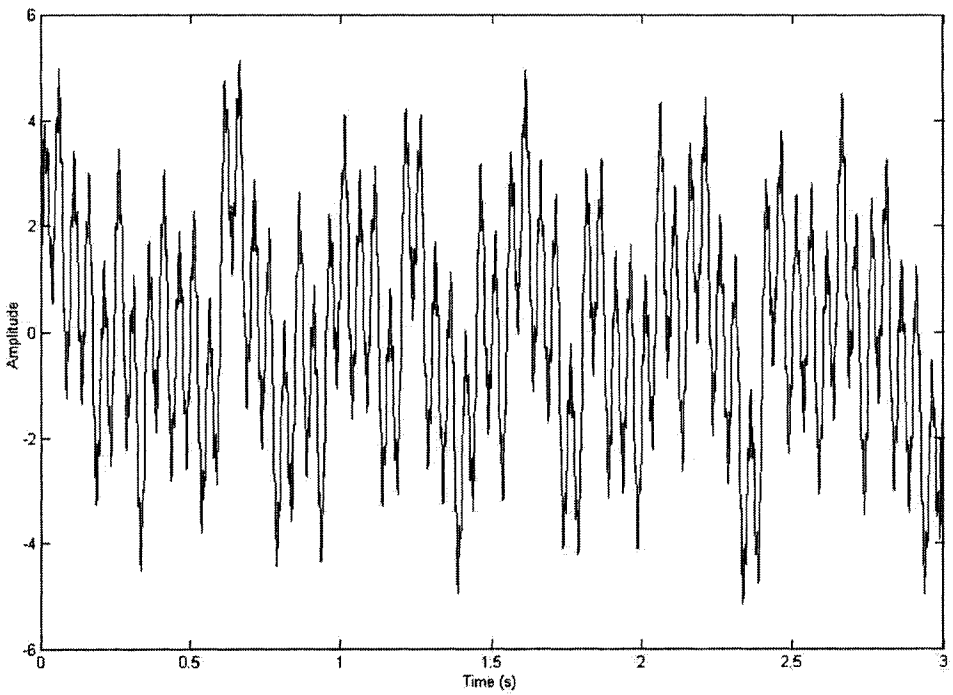

Figure 3.6: Combination of multiple sinusoid waves with different frequencies and amplitudes plus a global trend.

square error (MSE) between decomposed components and the sinusoid components. For this simulated signal, the MSEs between decomposed IMFs and correspondent frequency components are calculated as

$$MSE_i = \frac{1}{LOD} \sum_{j=1}^{LOD} [IMF_i(t_j) - x_i(_j)]^2, \quad i = 1, 2, \cdots, m, \tag{3.21}$$

where LOD is the length of the data, m is the number of known components, $IMF_i(t)$ is the ith IMF and $x_i(t)$ is the correspondent frequency component. A smaller i should represent a higher frequency component and a larger i should represent a lower frequency. The averaged MSE of the decomposition

Table 3.1: Comparison of stopping criteria on the 3000-point signal of multiple sinusoid waves plus trend.

Stopping Criteria	SD	S	Partial
CPU time	885.996s	0.216s	44.217s
MSE_{avg}	1214.90	N/A	695.38

is expressed as

$$MSE_{avg} = \frac{1}{m}\sum_{i=1}^{m} MSE_i. \tag{3.22}$$

To choose a proper stopping criterion, we use EMD algorithm with envelope means constructed by the cubic spline interpolation and without any consideration of end points swings. Three stopping criteria are compared, which are mentioned above. For the sake of convenience, we gave short names to them: SD (the standard deviation criterion in [27]), S (the criterion introduced in [26]) and Partial (the criterion introduced in [46]).

All codes are programmed in MATLAB 7.0 and have been tested on a Pentium 4 computer with 512MB of memory.

Table 3.1 shows the comparison on the 3000-point signal of multiple sinusoid waves plus trend. From Table 3.1 we can see that SD is the slowest criterion and the procedure takes a very long time to converge. S criterion is so easily met that it can be calculated very quickly, but only five IMFs are decomposed because some components are mixed in IMF3, IMF4, and IMF5 (Fig. 3.7) so the averaged MSE value can not be calculated. We don't like to see this serious mixture of frequency components in EMD. The partial criterion is a good tradeoff between SD and S.

Table 3.2 shows the comparison on the 8000-point signal of multiple sinusoid waves plus trend.

There are three points to be noted based on the comparisons above.

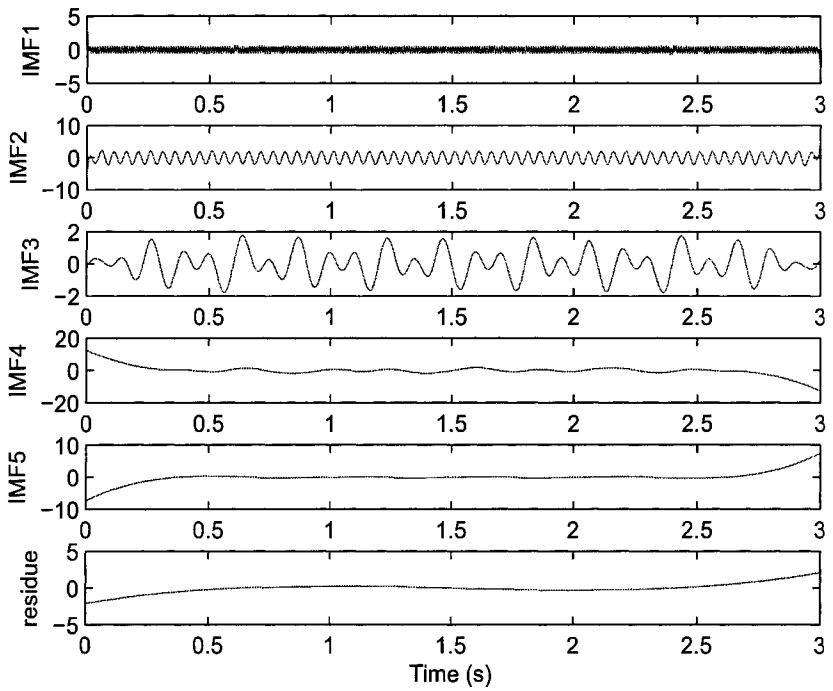

Figure 3.7: Decomposition of first type of simulated signal with S stopping criterion.

1. Some approaches to the end point swings are necessary because without them the averaged MSE would be very large, especially for short data.

2. Accuracy is improved as the length of data increases.

3. The partial criterion is a good tradeoff between SD and S with regard to CPU time and accuracy. We decided to choose partial criterion as the stopping criterion.

Then, we compared the extending approach and the mirroring approach with respect to the end point swing problem. In these comparisons, the partial stopping criterion and the cubic spline interpolation were used and the same

Table 3.2: Comparison of stopping criteria on the 8000-point signal of multiple sinusoid waves plus trend.

Stopping Criteria	SD	S	Partial
CPU time	40.866s	0.451s	89.042s
MSE_{avg}	162.65	N/A	390.89

Table 3.3: Comparison of approaches to the end point problem on the 3000-point signal of multiple sinusoid waves plus trend.

Approaches	Extending	Mirroring
CPU time	2.0310s	1.3440s
MSE_{avg}	0.2877	0.2414

short and long sinusoid signals were tested. Comparisons are shown in Table 3.3 and Table 3.4.

We can see that CPU time and accuracy are obviously improved by using approaches to the end point problem. From the comparisons, we found that the mirroring approach deals better with the end points than does the extending approach. As a result, we decide to use the mirroring approach to evaluate the proposed method with the original EMD.

3.2.3.2 Comparison of the Proposed Approach with the Reported Methods

Now, we can compare the proposed EMD that constructs envelope means by the modified monotone piecewise Hermite interpolation to EMD that constructs envelope means by cubic spline interpolation (SPLINE, in short) and by high-order spline interpolation (HIGH-ORDER, in short). Here the high-order interpolation is quartic, i.e. fourth order polynomial. With the help of Section 3.2.3.1, we selected the partial stopping criterion and the mirroring end point approach as the configuration to be used in testing the two methods under the same conditions. The first simulated signals used were a combination of multiple sinusoid waves, the same as those used in Section 3.2.3.1. These

Table 3.4: Comparison of approaches to the end point problem on the 8000-point signal of multiple sinusoid waves plus trend.

Approaches	Extending	Mirroring
CPU time	34.723s	5.383s
MSE_{avg}	111.3568	0.2768

signals gave us a basic assessment of each method's decomposition capability. All codes are programmed in MATLAB 7.0 and have been tested on the same Pentium 4 computer as mentioned. Figs. 3.8 – 3.10 show the decomposition of IMFs of the 3000-point data by the three methods. A primary assessment is given based on the visual observation of the IMFs. It can be seen from Figs. 3.8, 3.9 and 3.10 that:

- only high-order spline interpolation has different frequencies mixed up in an IMF (IMF4);
- MMPHI generates more IMFs than the other two interpolations;
- the first three sinusoidal components are decomposed in all figures unlike the rest components;
- although the cubic spline interpolation distorts the rest of the components a little worse than does MMPHI, the difference is not noticeable through observation.

These four points still hold for the decomposition of the 8000-point data. Thus, the averaged MSEs which were introduced in Section 3.2.3.1 are used on this type of signal as an objective indicator. The results are also shown in Tables 3.5 and 3.6. It can be seen that the MMPHI is always faster than HIGH-ORDER and is faster than SPLINE in the test of long data but is slightly slower than SPLINE in the test of short data. It can also be seend that the

Table 3.5: Comparison of the performance of the proposed method and other reported methods on the 3000-point signal of multiple sinusoid waves plus trend.

Interpolation methods	SPLINE	MMPHI	HIGH-ORDER
CPU time	1.3440s	1.4533s	1.5084s
MSE_{avg}	0.2414	0.1530	0.3126

Table 3.6: Comparison of the performance of the proposed method and other reported methods on the 8000-point signal of multiple sinusoid waves plus trend.

Interpolation methods	SPLINE	MMPHI	HIGH-ORDER
CPU time	5.3830s	3.7443s	4.3187s
MSE_{avg}	0.2768	0.0846	0.3266

proposed MMPHI approach is better than the cubic spline and the high-order spline interpolations in terms of the averaged MSE, with an improvement of up to 74.3%.

The second type of simulated signal is a combination of a periodic impulse signal and a chirp signal. Each impulse can be expressed as

$$y_i(t) = 0.1e^{-100t}sin(1000t). \tag{3.23}$$

The time interval between every two impulses is 0.25 seconds. The chirp signal can be expressed

$$y_c(t) = sin(100\pi t^2). \tag{3.24}$$

The signal is shown in Fig. 3.11. The mixing ratio between the impulses and the chirp signal is 1:1. From this figure, we can see that the impulses are hard to identify in the mixed signal. Because periodic impulses often represent fault signatures of machinery, this simulated signal is used to test each method's ability to detect the existence of periodic impulses. We also test both a short series (3000 points) and a long series (8000 points) and set the time interval of

Table 3.7: Comparison of the performance of the proposed method and other reported methods on the 3000-point signal of combination of impulse and chirp.

Interpolation methods	SPLINE	MMPHI	HIGH-ORDER
CPU time	10.073s	3.6877s	264.6530s
MSE_{avg}	0.0338	0.0099	0.1080

Table 3.8: Comparison of the performance of the proposed method and other reported methods on the 8000-point signal of combination of impulse and chirp.

Interpolation methods	SPLINE	MMPHI	HIGH-ORDER
CPU time	7.9107s	7.7233s	550.3680s
MSE_{avg}	0.0012	0.0012	0.0044

any two adjacent points at 0.001s. Figs. 3.12 – 3.14 show the decomposition of IMFs of the 3000-point data by the three methods. From Figs. 3.12, 3.13 and 3.14, it can be seen that there is not much difference based on visual observation. The chirp signal stays mainly in the first IMFs and none of the three figures gives clear information of the impulses. The decomposition of the long signal gives the same observation. As a result, the averaged MSE is still used as an indicator of accuracy but this time $m = 2$ in equation (3.21). The results are shown also in Tables 3.7 and 3.8. The proposed MMPHI approach is the best with regard to CPU time and accuracy. The HIGH-ORDER is very slow. In the test of the long signal, there was not as much difference between SPLINE and MMPHI as that of the short signal.

3.3 A Test on Experimental Data

Through comparisons on simulated signals, we have established the superiority of the approximation that uses the modified monotone piecewise Hermite interpolation. Now we will look at how it performs on an experimental data. It has been applied to the set of data from the gearbox experiment that was described on page 21 and the decomposition result is shown in Fig. 3.15. The

MMPHI takes 25.935 seconds to complete the decomposition. The first IMF obtained enlarged in Fig. 3.16. Compared with the original collected data in Fig. 2.5 on page 23, we can see that the obtained first IMF shows impulses much more clearly than do the original data. It is easy to measure the distance between two impulses which is 0.18 seconds, and represents the frequency of the output shaft ($1/0.18 = 5.5$ Hz). This is what we would expect to see because the faulty gear is mounted on the output shaft.

3.4 Summary of This Chapter

This chapter has reviewed methods of approximating the envelope-mean in EMD, focusing on interpolation methods reported in the literature. The advantages and disadvantages of these methods have been presented, as well. The modified monotone piecewise Hermite interpolation was used as a replacement for the cubic spline and high-order spline interpolations. Thereafter, a test was conducted to select parameters to be used in verifying our expectations of improvement, specifically, in selecting stopping criteria and approach to dealing with end point swings. Once that was done, the three methods were compared. It can be seen from that comparison that the proposed approach is in most cases better than the reported envelope-mean methods with respect to CPU time and accuracy. It should also be noted that even though the proposed method has the smallest value of averaged MSE on the chirp plus impulses signal, in Figs. 3.13 - 3.14, none of the methods has identified the impulses clearly in time domain. Although we can see some impulses from the experimental test, more impulses are expected to be identified clearly. Improvement on this point will be taken into account in Chapter 4 where a direct-mean approximation will be used.

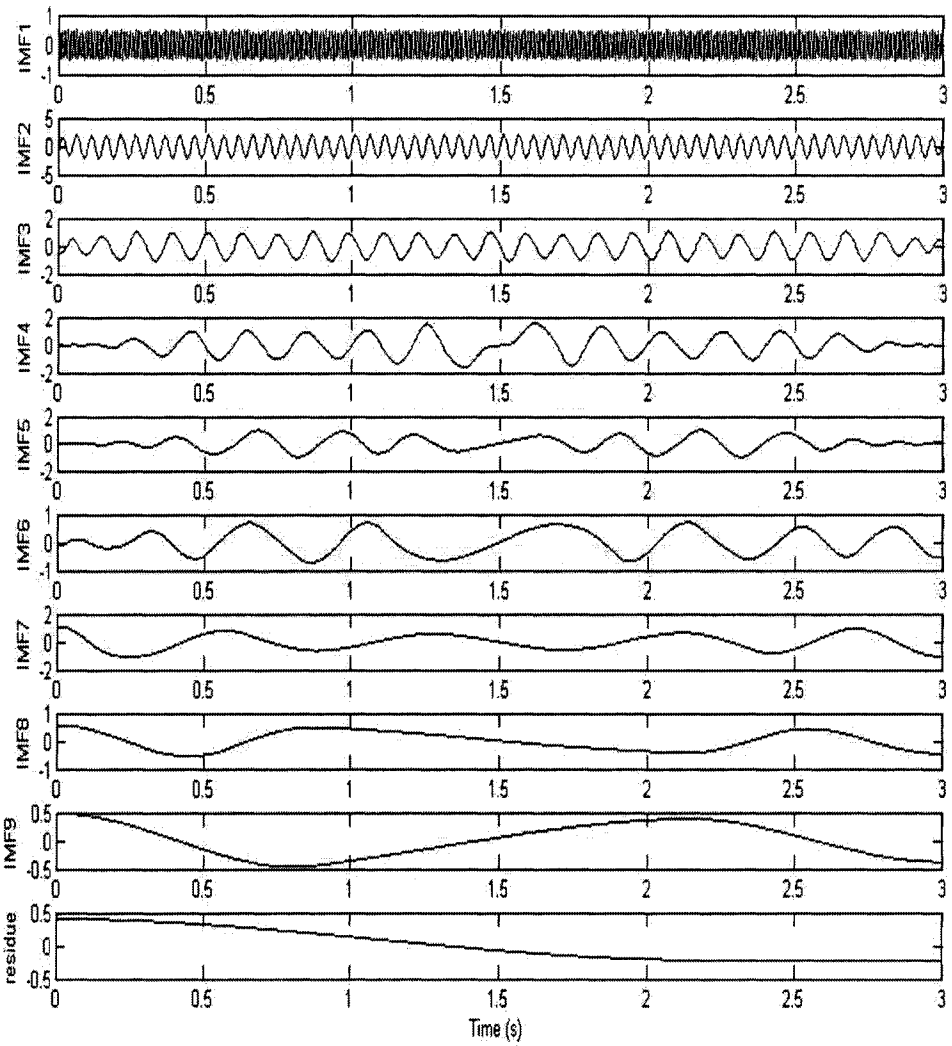

Figure 3.8: Decomposition of the signal of multiple sinusoid waves plus trend with MMPHI (3000 points).

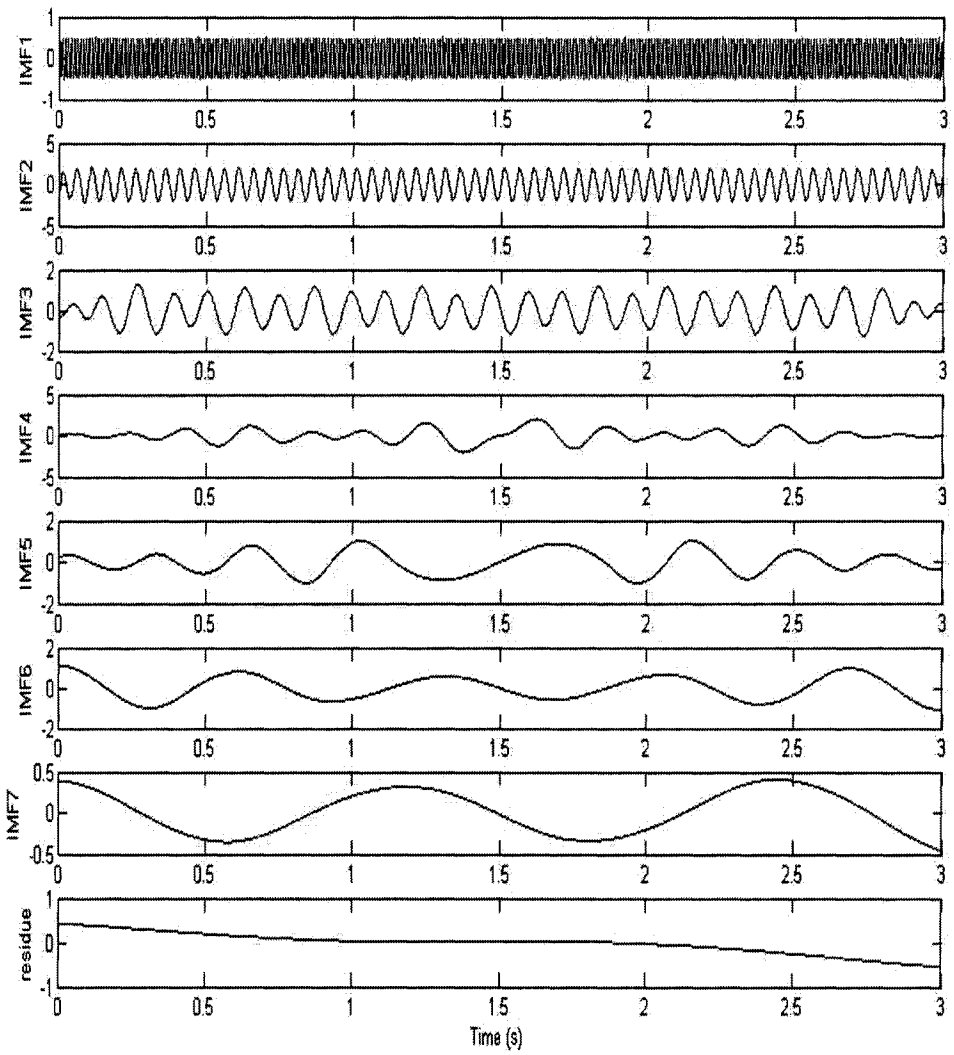

Figure 3.9: Decomposition of the signal of multiple sinusoid waves plus trend with the cubic spline interpolation (3000 points).

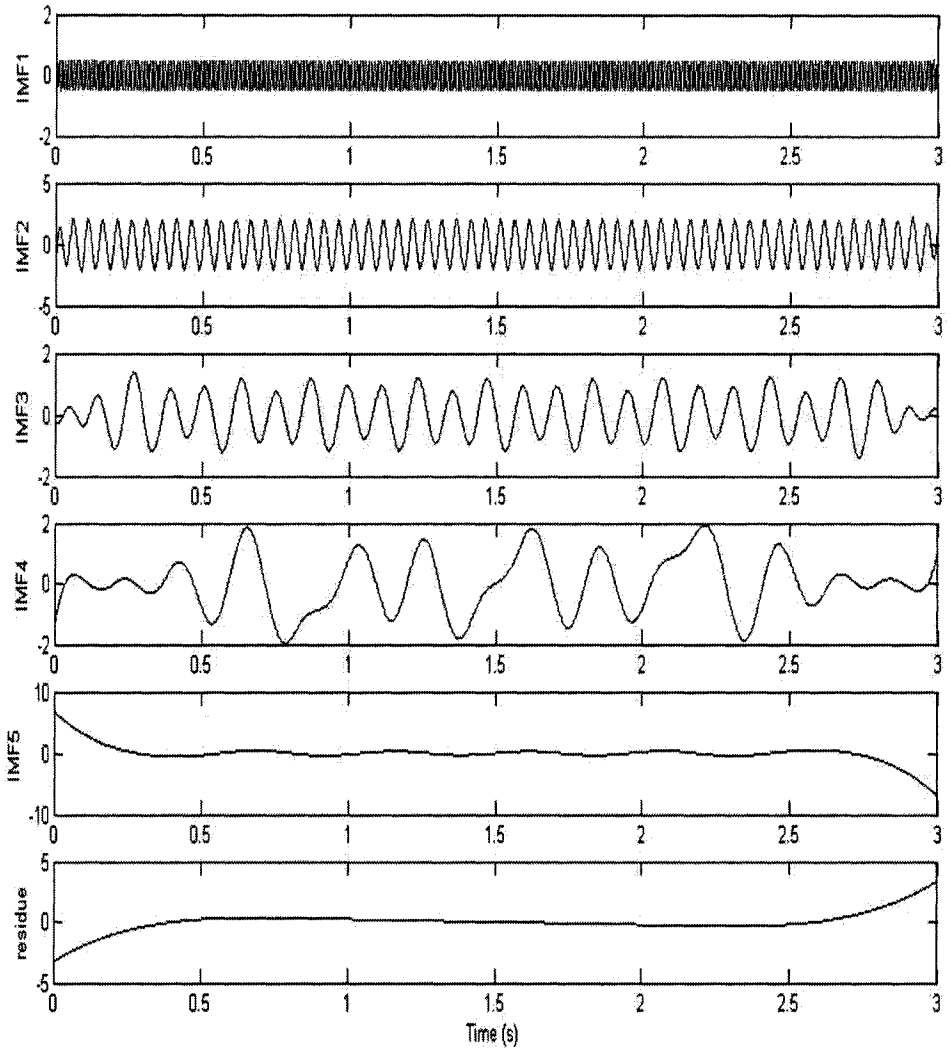

Figure 3.10: Decomposition of the signal of multiple sinusoid waves plus trend with the high-order spline interpolation (3000 points).

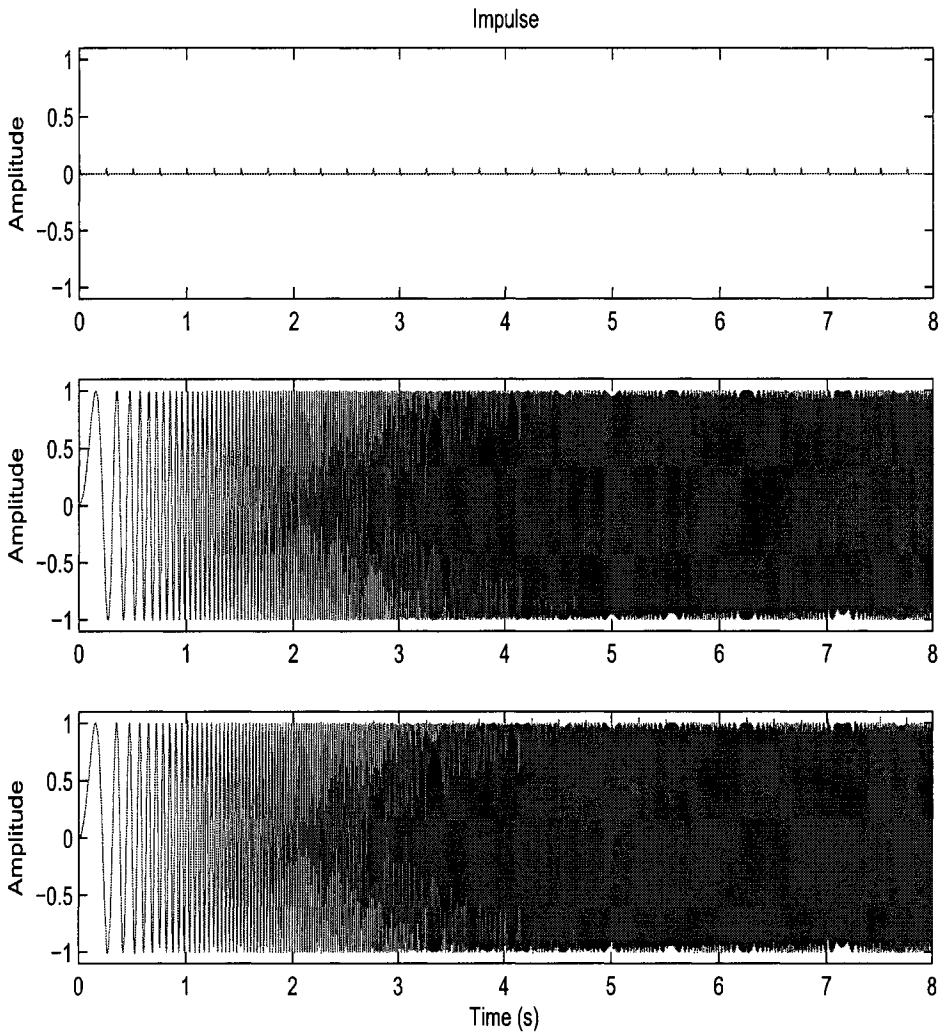

Figure 3.11: Combination of a periodic impulse signal and a chirp signal.

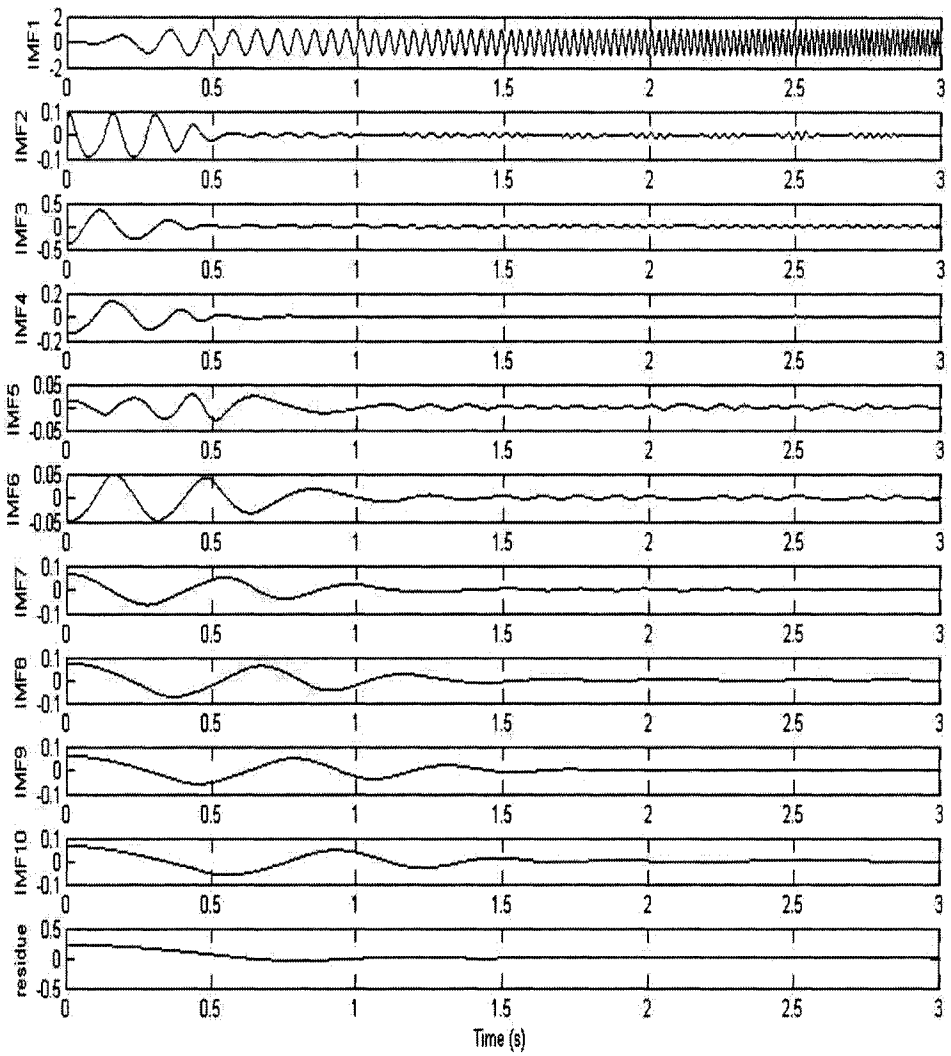

Figure 3.12: Decomposition of a combination of impulse and chirp with the MMPHI (3000 points).

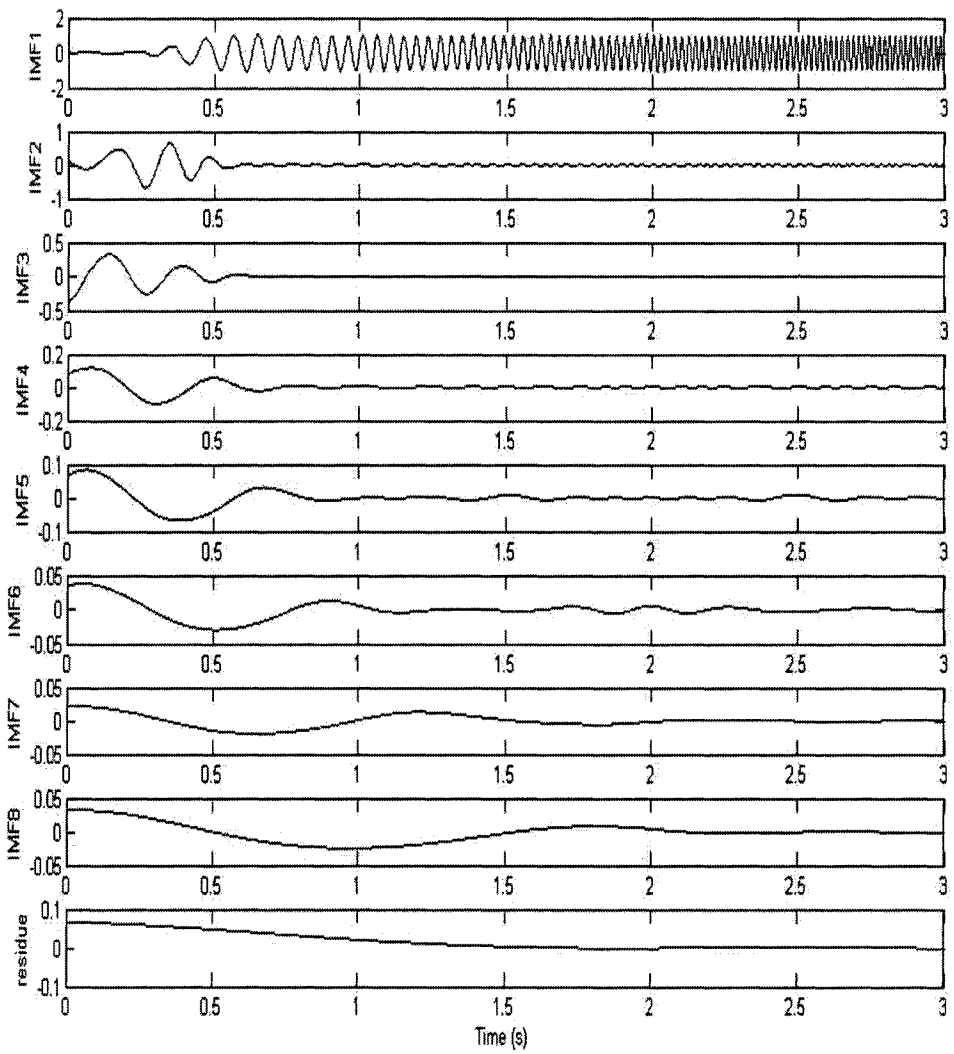

Figure 3.13: Decomposition of a combination of impulse and chirp with the cubic spline interpolation (3000 points).

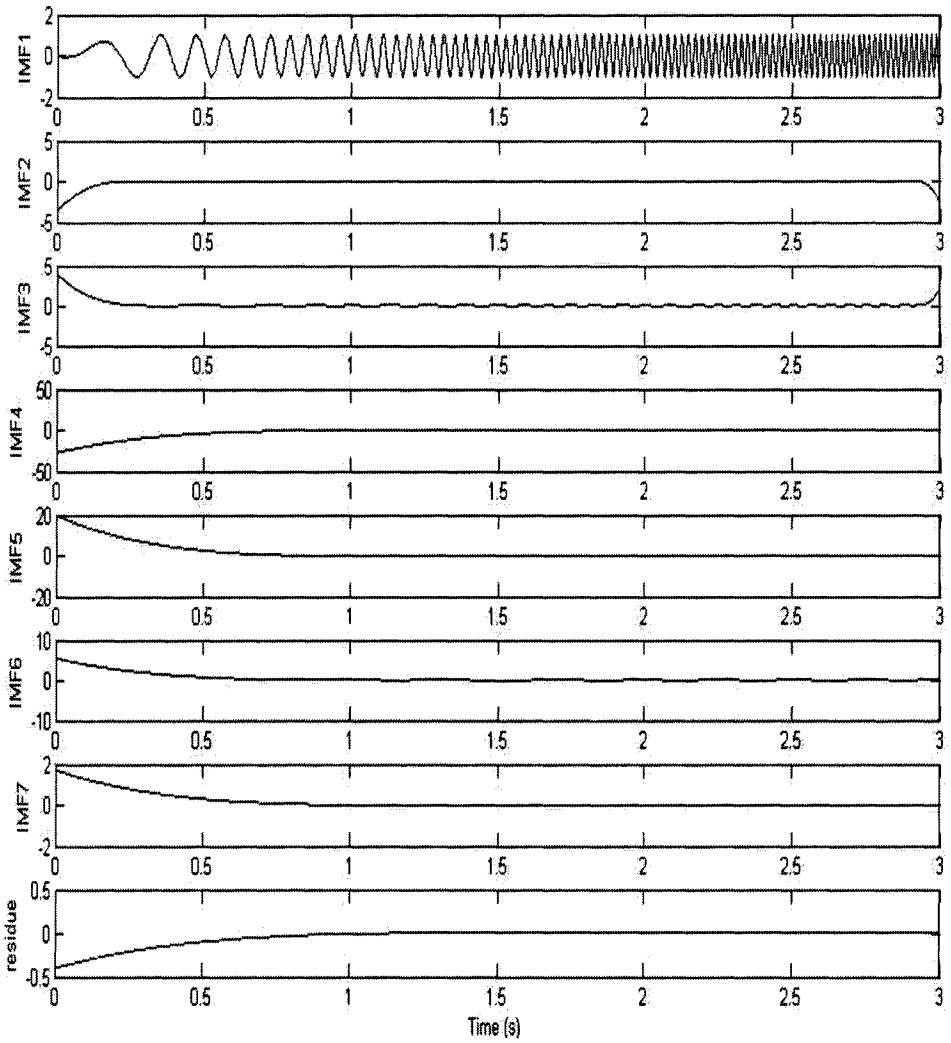

Figure 3.14: Decomposition of a combination of impulse and chirp with the high-order interpolation (3000 points).

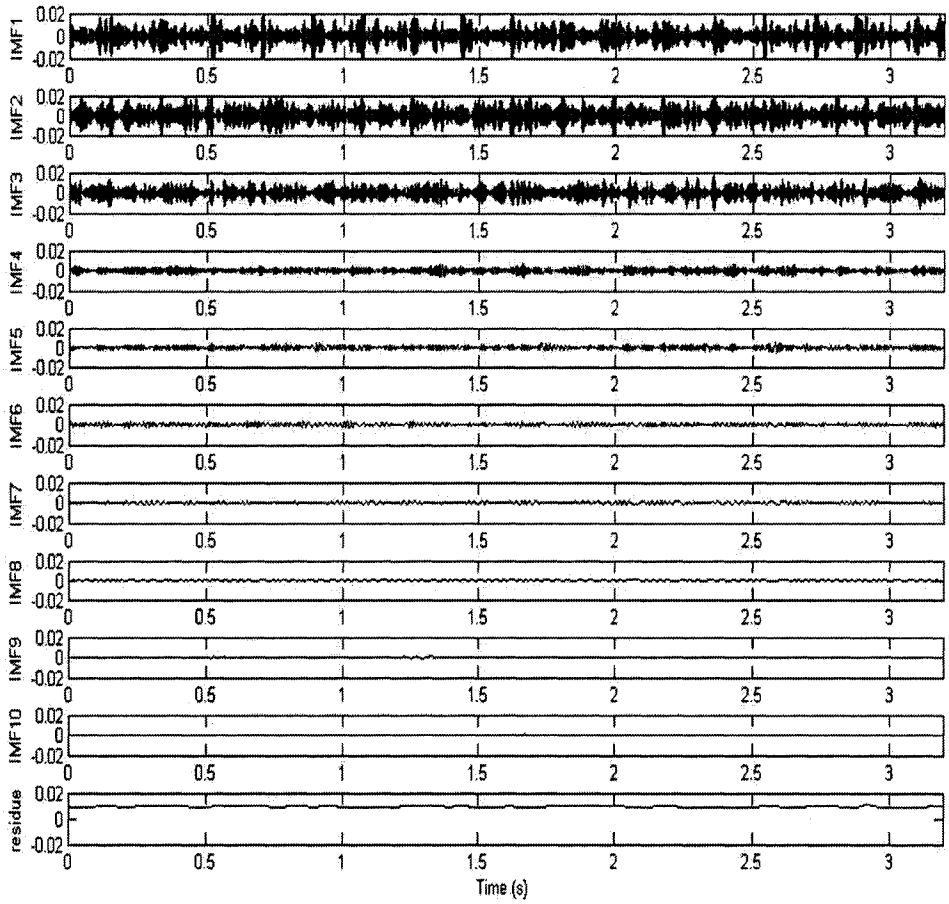

Figure 3.15: Decomposition of the vibration data set using EMD with MMPHI.

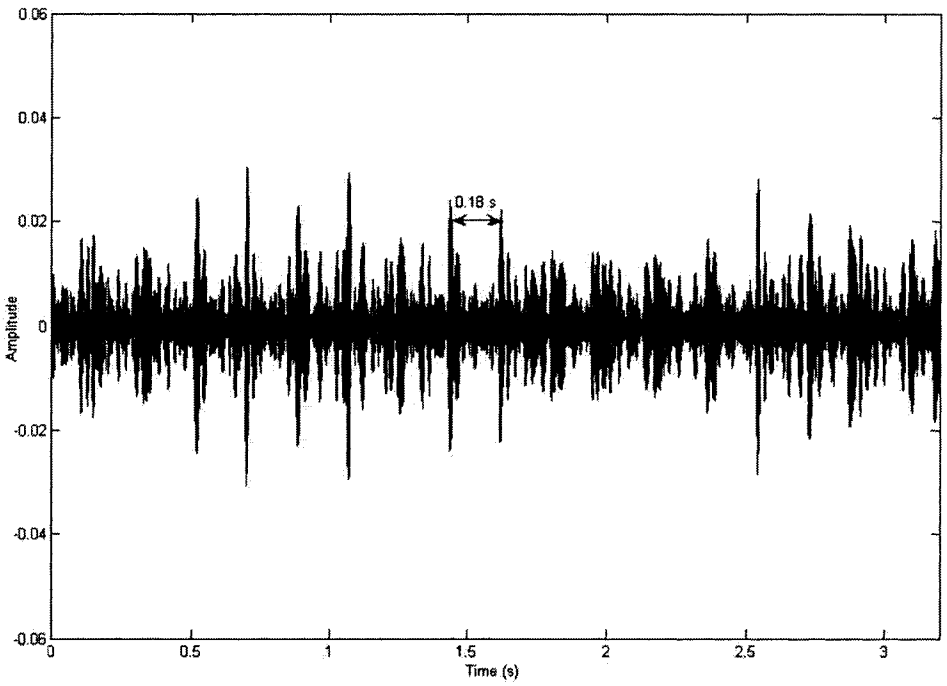

Figure 3.16: The first IMF in Fig. 3.15.

Chapter 4

Improvement of the Direct Approximation of the Local Mean

In the sifting process of the original EMD method introduced in Chapter 2, the envelope mean is calculated as an approximation to the local mean. In Chapter 3, we introduced the improvement to the envelope-mean approximation by using monotone piecewise Hermite interpolation. At the end of Chapter 3, however, we pointed out a common problem of all the three interpolation approaches used for the envelope-mean approximation in EMD–that impulses cannot be identified clearly in decomposition. This is due to a shortcoming of envelope-mean methods. Essentially, they all tend to use envelope means to approximate local means. The phrase "local mean" means that there are two requirements for the calculated curve: 1) it should be the mean of a signal so it should reflect the global trend of the signal, and 2) some regional signatures of the original signal should appear on the curve of the local mean. These two requirements guarantee that the difference between the original signal and the local mean converges on an IMF. But the algorithm for calculating envelope means is to connect maxima and minima of the wave by a given interpolation method and obtain the mean of the upper and lower envelopes. Values for

the points between extrema are really not important as long as they don't exceed their neighboring extrema and become new extrema; therefore, for two signals with the same extrema positions and values but quite different points between the extrema, the means obtained by the envelope-mean approach will be same. This is proof that envelope means do not approximate very well to local means.

Thus, in this chapter, we work on an alternative to the envelope-mean approximation to the local mean. This type of approximation is called "direct approximation". Reported methods of this type are reviewed in Section 4.1. An integrated windowed local mean method is proposed and comparison between it and another direct approximation is presented in Section 4.2. A discussion follows in Section 4.3 and the performance of the winner of the comparison on an experimental data is given in Section 4.4.

4.1 Reported Direct Approximation Methods

4.1.1 Local Mean Mode Decomposition

Gai *et al* introduced a method for approximating local means called Local Mean Mode Decomposition (LMMD) [21]. LMMD uses a time-varying filter to calculate the means. First, all the local extrema, $Z(t_i)$ (i is the index of the ith extremum), of the signal $x(t)$ are found. Then the mean value, $e(i)$, between two successive local extrema, $Z(t_i)$, $Z(t_{i+1})$, can be found using the equations below in which t_i means the time value of the ith local extremum. There are two or more time spots between two extrema. Finally, to get the "local mean", LMMD uses

$$m(t_{i+1}) = h(t_i) \times e(i) + h(t_{i+1}) \times e(i+1), \tag{4.1}$$

where

$$\begin{aligned} h(t_i) &= \frac{t_{i+2} - t_{i+1}}{t_{i+2} - t_i}, \\ h(t_{i+1}) &= \frac{t_{i+1} - t_i}{t_{i+2} - t_i}, \\ e(i) &= \frac{1}{t_{i+1} - t_i + 1} \sum_{l=t_i}^{t_{i+1}} x(l), \\ e(i+1) &= \frac{1}{t_{i+2} - t_{i+1} + 1} \sum_{l=t_{i+1}}^{t_{i+2}} x(l). \end{aligned}$$

After obtaining $m(t_1), \cdots, m(t_n)$ (n is the number of extrema), the cubic spline interpolation is used to connect them as m_{11} in equation (2.14). The other steps are the same as theses of EMD. If, however, the data being treated is non-stationary, the local mean involves a local time scale for computing the mean; this time scale is impossible to define [27]. As a result, the "local mean" as it is called in [21] is not a real local mean because the time interval between two extrema is not local time scale. LMMD's using the cubic spline interpolation would result in the same shortcoming encountered with the envelope-mean approximation and no effort was done to support the statement that LMMD is much faster than EMD.

4.1.2 The Windowed Local Mean

Rösler [45] considered the mean of a signal within a window and called this the local mean with respect to that window. In this thesis, this type of direct approximation is termed a "windowed local mean". For a continuous function, $y = f(x)$, the expression of its windowed local mean is

$$m_\delta[f](x) = \frac{1}{\delta} \int_{x_\delta} f(\tau) d\tau, \tag{4.2}$$

where $x_\delta = [x - \delta/2, x + \delta/2]$, and δ is the width of the integration window. The purpose of giving such an expression in [45], however, is not for using it for EMD but just to introduce the windowed local mean to interpret the concept of "local mean" in Huang's paper [27]. Only an example on how to find the right window width for a continuous signal of sinusoid components was given by [45] but no integrated algorithm was provided that could be applied in EMD.

4.2 Proposed Direct Approximation Using the Windowed Local Mean

4.2.1 The Motivation and the Expectation

The purpose of this chapter is to find an improved direct approximation with a better capacity for identifying impulses. Such an approximation should consider the relatively macro view of a signal but the consideration has to be restricted to within relatively micro windows to avoid losing local signatures. Although Rösler [45] mentioned only one example of a function of the windowed local mean, we can see that the method does have this good feature. An expression of the windowed local mean for discrete data points is not given in [45] but is needed to deal with real signals and to process them in a computer. We define that for a set of data, x_i, $i = 1, 2, \cdots, n$, the discrete form of its windowed local mean is

$$m_\delta(i) = \frac{1}{\delta + 1} \sum_{j=i-\delta/2}^{i+\delta/2} x_j, \quad i = 1, 2, \cdots, n, \tag{4.3}$$

where $\delta+1$ is the number of data points in the window centered at data point x_i. It's apparent that δ has to be an even integer. The LMMD method reviewed in

Chapter 3 [21] also calculates windowed local means using the distance between two consecutive extrema as the width of a summation window. But from equation (4.2) we can see that only one representative mean value is computed for each pair of two consecutive extrema and the final mean curve is obtained by connecting these mean values with the cubic spline interpolation. Using the method proposed in this chapter, windowed local means are calculated at each point of the data set. The width of a summation window is centered at this point. Thus, all points are utilized to contribute to the approximation of the local mean and no interpolation is required.

Thus, we propose to apply the discrete form of the windowed local mean to EMD to directly approximate the local mean. We expect it to have a better capacity for identifying impulses without sacrificing too much basic decomposition capacity and adding too much CPU time.

To have an integrated algorithm, two issues need to be discussed in next subsections: selection of a window width and end point extension.

4.2.2 Selection of the Window Width

Apparently, selection of the proper width of the summation window is absolutely crucial to the effectiveness of the windowed local mean method because it determines the relative relationship between the local and global perspectives. If the width is equal to the length of the data set, all the mean values are the same and are equal to the algebraic mean of the data set. As a result, the calculated mean will be a flat curve. If the original data has a non-zero mean, it will be moved vertically after the subtraction of the flat mean curve so it will be symmetrical about the X-axis. If the original data has a zero mean already, it will not be changed after the subtraction. On the other hand, if the width

is compressed to 0, the mean values are just the points of the original data themselves so the mean curve is the signal itself and the rest of the subtraction is the X-axis. Clearly, a proper width should be selected between these two extreme cases.

Let's look at a simple example, a sinusoid signal with a single frequency component, for instance, $x(n) = sin(2\pi n/100)$, $n = 1, 2, \cdots, 1000$. Apparently, an ideal decomposition of it should produce itself as the IMF and zero mean as the residue. This requires the windowed local means to be zero. It will be seen that if the period or multiples of the signal period are used as the width of the window, this requirement will be fulfilled. As a result, the shortest feasible window width is the period of the sinusoid signal.

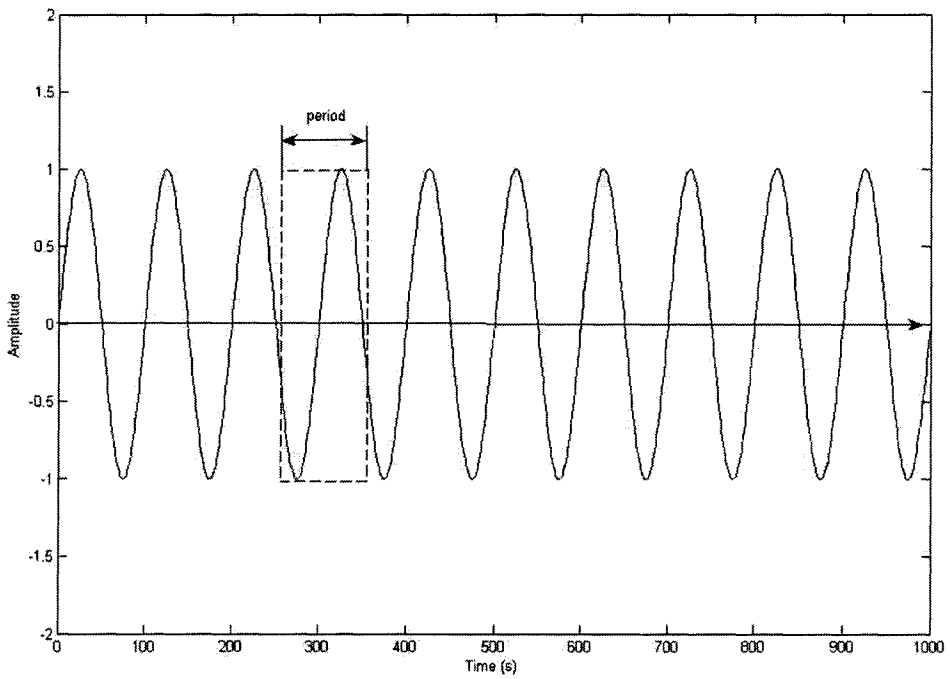

Figure 4.1: Illustration of a window width for a sinusoidal signal. The region means the window has a width equal to its period.

Let's take an example to see how to find a feasible width for a signal with multiple frequency components. We have a combination signal with three frequency components and a length of 3000, This is expressed as

$$x(n) = sin(2\pi n/1000) + sin(2\pi n/500) + \frac{1}{2}sin(2\pi n/100), \quad n = 1, 2, \cdots, 3000. \tag{4.4}$$

Periods of the three components are 1000, 500, and 100, respectively. If we select a shorter window width, e.g. 50, which is less than the shortest period of the three components, the result of applying the windowed local mean method to the combination signal to decompose the first IMF is as shown in Fig. 4.2. It can be seen from Fig. 4.2 that although the highest frequency component is filtered out in the subtraction (the first IMF) from the original signal, it still exists in the windowed local mean curve. This means that as the mean is used as a new signal for the next round of decomposition, it is not free of the highest frequency component so the decomposition is inefficient.

If we select a longer width, e.g. 500, which is the period of a lower frequency component, the result of applying windowed local mean method to the combination signal to decompose the first IMF is as shown in Fig. 4.3. It can be seen from Fig. 4.3 that although the windowed mean captures a general trend of the signal (period of 1000), the difference between the signal and the mean is still mixed with a low frequency component (period of 500). It cannot be used as an IMF. If we select a width between 100 and 500, e.g. 300, it can be seen from Fig. 4.4 that the decomposition is even worse than in the case of 500, that both the window local mean curve and the first IMF curve are mixtures of frequencies.

If a width is selected to be exactly equal to the shortest period, or the

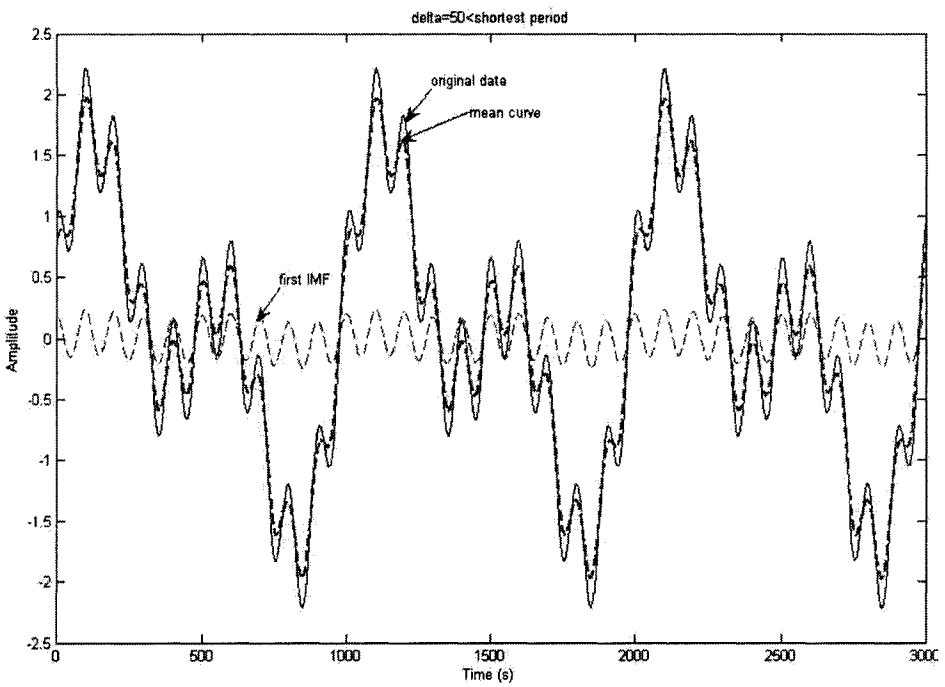

Figure 4.2: Decomposition of the signal's first IMF using $\delta = 50$, which is less than the shortest period. The dotted line is the original signal; the solid line is the windowed local mean; the dashed-and-dotted line is the difference between the signal and the windowed local mean.

period of the highest frequency component, i.e. 100, it can be seen from Fig. 4.5 that the difference between the signal and the windowed mean contains only the highest frequency and the windowed local mean is built up only by the lower frequencies. As a result, the thin dashed line represents the first IMF and only lower frequency components need to be further decomposed in following IMFs. It can be concluded from our tests on optional widths that, for this type of signal with multiple frequency components, a good trade-off between global and local requirements would have the width of the windowed local mean equal the period of the highest frequency component. It is noted that the frequencies of this example are multiples of one another. We believe

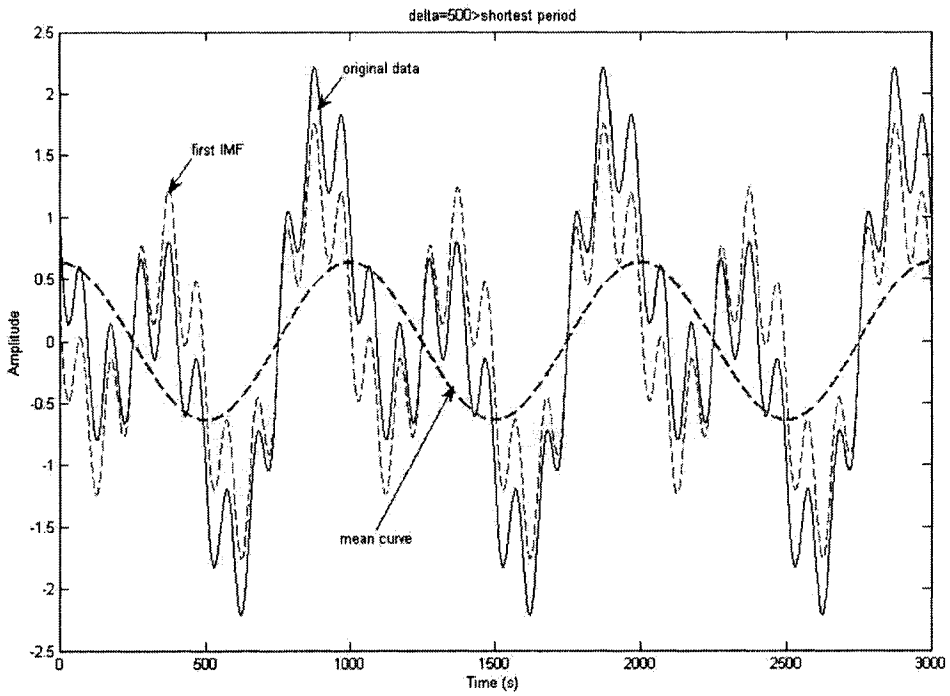

Figure 4.3: Decomposition of the signal's first IMF using $\delta = 500$, which is greater than the shortest period.

a same conclusion can be reached if frequencies are randomly selected since we don't utilize any benefits of being multiples. But it needs to be verified by testing more types of signals. This is an empirical result since we have not tested all widths from 1 to 3000; to do so would be too time-consuming. Rösler [45] used a similar example but made a mistake in saying that the proper width is half of the period of the highest frequency component.

For a given signal, the period of the highest frequency component is difficult to determine unless we are clear about its physical mechanism. For example, for the signal shown in Fig. 2.5 on page 23, the procedure of using a given period as the width cannot work because no period is known. Alternatively, we can use the interval between two neighboring maxima or two neighboring

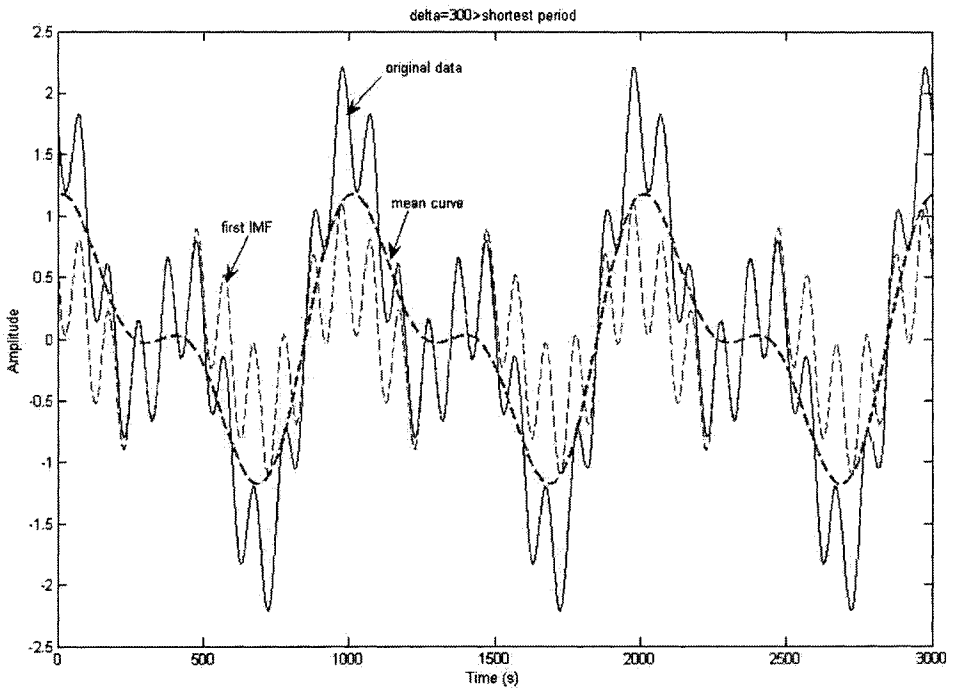

Figure 4.4: Decomposition of the signal's first IMF using $\delta = 300$.

minima to estimate the shortest period. If the values of the shortest periods are not the same, we can calculate the average length of all intervals between two neighboring maxima and between two neighboring minima first and then use this average value to calculate windowed local mean at the current level. When the IMF at this level is found, the process moves on to the next level and requires us to calculate an average value again for the next IMF decomposition. The decomposition process will not stop until the residual is a trend signal or the number of the decomposed IMFs has reached a pre-set value. This procedure works for the signal in Fig. 2.5 and the result is shown later in Fig. 4.18 in Chapter 5.

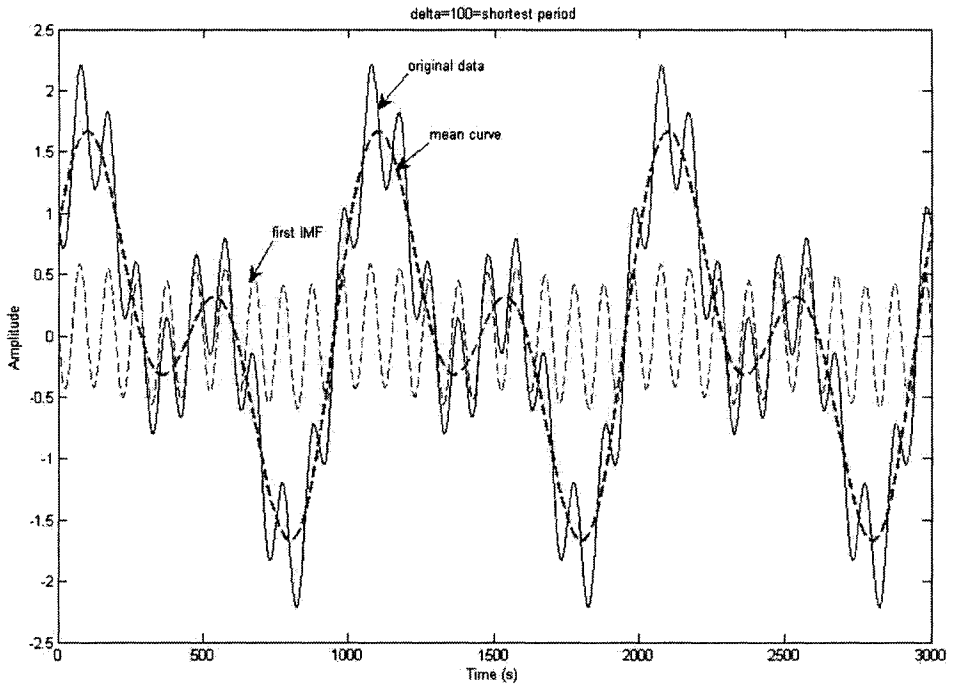

Figure 4.5: Decomposition of the signal's first IMF using $\delta = 100$, which is equal to the shortest period.

4.2.3 End Point Extension

In the last subsection, we introduced the concept of the windowed local mean and gave a method to find a proper width for it. The procedure for a decomposition using the windowed local mean is also described. In this subsection, we will point out a problem regarding the windowed local mean and give a solution to the problem.

As shown by equation (4.3), the windowed local mean is calculated as a summation at each data point. The summation range is centered at the data point and has a width which is a given value at a certain decomposition level. The calculation is not a problem as long as these data values are available for calculating summations. When the distance between a center point and its

nearest end point is less than half the width of the window, however, no more data values can be provided beyond the end point and the summation cannot be completed. Especially, for the first and last points, only half of the data values are available. Thus, we have to have the ends of the data extended to guarantee that a summation can be conducted at every data point. The data should be extended at least half of the window width at the first and last point. Since we don't have any idea what is happening beyond the time span for data collection, we don't like to impose any artificial assumptions on a signal without strong evidence. We choose sine waves for extending a signal because they have explicit expressions. When we are extending the end points of a signal in an envelope-mean approximation, we care about only the positions and values of extrema. Here, we have to consider values of every extended point since all of them are used in the calculation of the windowed local mean. With an explicit expression, we know the values of as many extended points as we want so we don't need to specify so many points. A detailed description of the extension is explained in Fig. 4.6. The solid curve is the end part of an arbitrarily given signal. Only the extension of last points is shown because the extension of first points is similar. In this example, the last extremum is a maximum and the value of the last point (V_P) is greater than the value of the last minimum (V_{min}). We extend the signal with a sine wave, as

$$f(t) = A\cos(\frac{2\pi t}{period} + phase) + B, \tag{4.5}$$

where A is the amplitude determined by the values of the last minimum and the last maximum: $A = \frac{1}{2}|V_{max} - V_{min}|$, $period$ is determined by the interval between the last two extrema: $period = 2 \times |t_{max} - t_{min}|$, B is the vertical

movement that makes the sine wave symmetric about the t'-axis ($B = \frac{1}{2}[V_{max} - A]$ in this example), *phase* is the phase of the sine wave to slide along the t'-axis until it has the same value (V_P) as the last data point, P, at the last time spot t_n ($phase = arccos[(V_p - B)/A]$ in this example). These parameters make the extended wave continuous with the end of the data on value, trend and boundary. If the value of the last point (V_P) is less than the value of the last minimum (V_{min}), the only change to the extension is that the boundary of the sine wave is determined by the last point and the last maximum, i.e. $A = \frac{1}{2}|V_{max} - V_P|$. If the last extremum is a minimum, the extension method is analogous to the case in which the last extremum is a maximum.

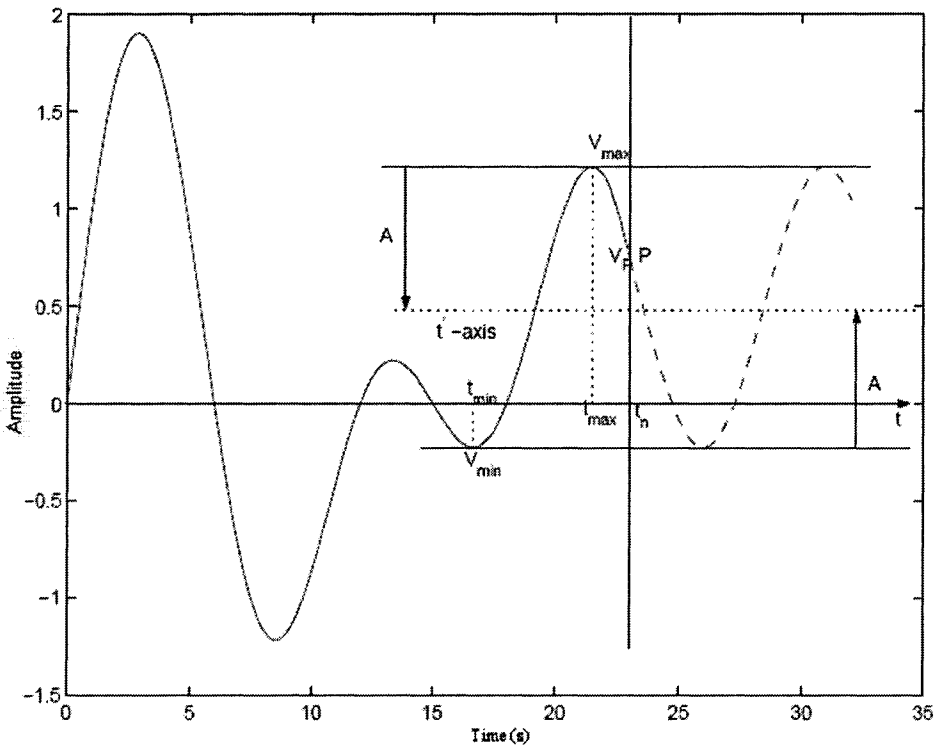

Figure 4.6: Illustration of the extension of the end of a signal.

Fig. 4.7 shows the result of the windowed local mean approximation when the extension procedure is applied to the signal in equation (4.5). There are not any obvious abnormal curves introduced to the mean curve and the IMF curve. The extension procedure is feasible for this signal.

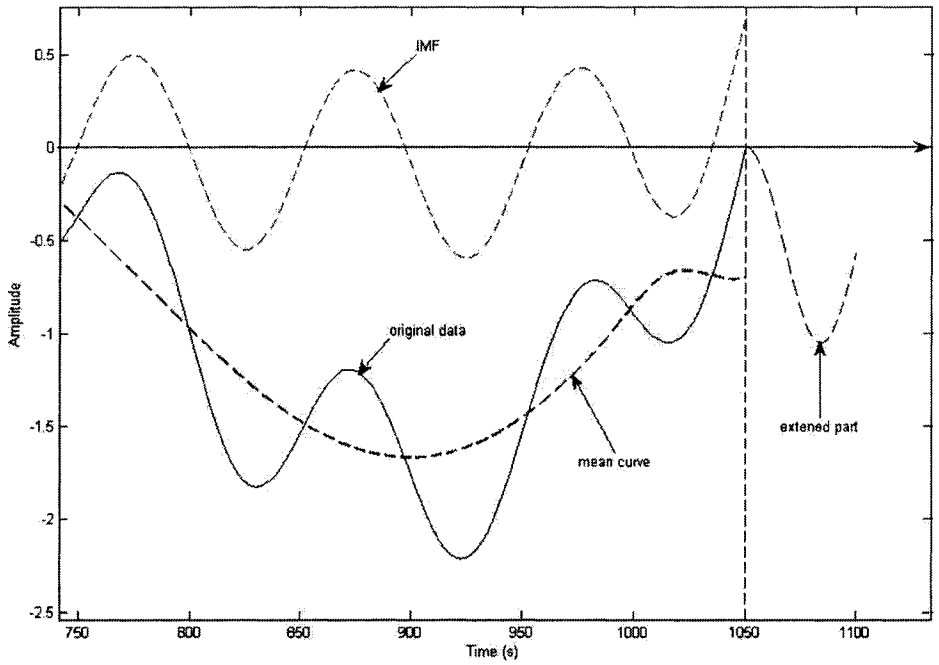

Figure 4.7: Result of the windowed local mean approximation after the extension of the end points.

4.2.4 Comparisons on Simulated data

In this section, we are going to use the same simulated signals used in Section 3.3 to test the proposed windowed local mean method and compare it to the method which uses the reported direct-mean approximation (LMMD [21]). The proposed method uses the procedures in Sections 4.2.2 and 4.2.3 to select a window width and to extend the end points. For the sake of convenience,

the two signals are shown again below.

- Combination of multiple sinusoid waves with different frequencies and amplitudes plus a global trend:

$$\begin{aligned} y(t) = \; & 0.5sin(2\pi * 1000t/10) + 2sin(2\pi * 1000t/50) + \\ & sin(2\pi * 1000t/120) + sin(2\pi * 1000t/200) + \\ & sin(2\pi * 1000t/300) + sin(2\pi * 1000t/500) + 0.0005t. \end{aligned} \tag{4.6}$$

- Combination of a periodic impulse signal and a chirp signal:

$$y(t) = y_c(t) + y_i(t), \tag{4.7}$$

where $y_c(t) = sin(100\pi t^2)$ and $y_i(t) = 0.1e^{-100t}sin(1000t)$.

The comparison on the combination of sinusoid waves is actually an assessment of a method's ability to decompose standard stationary signals. The CPU times for the proposed approach and the LMMD on the short (3000 points) and long (8000 points) multiple sinusoid combination are shown in Tables 4.1 and 4.2. We can see that the proposed method does not have an advantage with regard to CPU time. This is because at every level of decomposition, every data point participates in the calculation of the windowed summation. As the number of extrema decreases with decomposition, the width of the summation window becomes larger and larger so that time consumption increases as the decomposition nears its conclusion, with regard to decomposition performance, however, we will show that the proposed approach out-performs LMMD. From observing Figs. 4.8 and 4.9, it can be seen that there is a serious problem with LMMD. Only four IMFs have been generated because components are mixed up. As is evident in Figs. 4.10 and 4.11 the proposed approach does not have

Table 4.1: Comparison of the CPU times for the windowed local mean approximation and the LMMD method on a 3000-point singal of multiple sinusoid combination.

Direct-mean approximation	LMMD	WINDOWED
CPU time	0.1400s	4.0800s

Table 4.2: Comparison of the CPU times for the windowed local mean approximation and the LMMD method on an 8000-point signal of multiple sinusoid combination.

Direct-mean approximation	LMMD	WINDOWED
CPU time	2.9700s	16.5070s

this problem. Although the IMFs beyond IMF3 are distorted to some extent, at least the frequencies of the first three IMFs are clear and equal to the designed components.

We applied the two methods to the chirp and impulse signals and the resulting CPU times are shown in Tables 4.3 and 4.4. The proposed approach is slower than LMMD for the reason mentioned above but the time consumption is not that large. From visual observations of the decompositions recorded in Fig. 4.12, it can be seenthat some impulses are present in IMF4-IMF10. The marked region in IMF6 is enlarged in Fig. 4.13 where most of the periodical impulses are clearly shown and it is easy to identify their time interval is just 0.25 seconds as designed. The appearance of impulses is even more obvious when the proposed approach is applied to long combination of impulse and chirp. IMF4-IMF10 all consist of the impulse patterns shown in Fig. 4.14, and the enlarged marked region of IMF7 shows up as the clearest one in Fig. 4.15. In this figure, impulses are separated clearly and the shape of them is close to the original impulses that are shown in Fig. 3.11. In Figs. 4.16 and 4.17, however, the information on impulses is too blurred to reveal their intervals even from enlarged IMFs, which is our real concern when this type of simulated

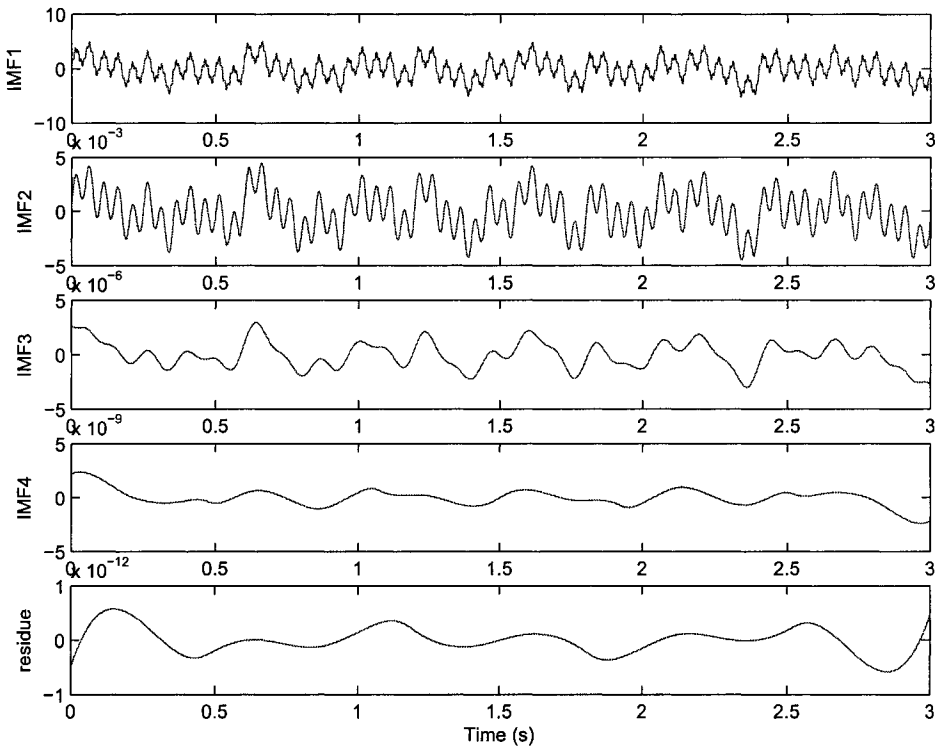

Figure 4.8: Decomposition of a 3000-point sinusoid combination using LMMD.

signal is tested, because impulses may represent faults being monitored.

4.3 Discussion of the Comparisons

This chapter introduced an integrated procedure for using the windowed local mean as a proposed direct-mean approximation. This procedure includes the selection of proper window widths and extension of end points. Based on the comparison with the LMMD method, the proposed approach is not as fast as LMMD because the width of the summation window becomes larger and larger causing time consumption to increase as the decomposition approaches its conclusion. The LMMD method, however, does not pass the basic decom-

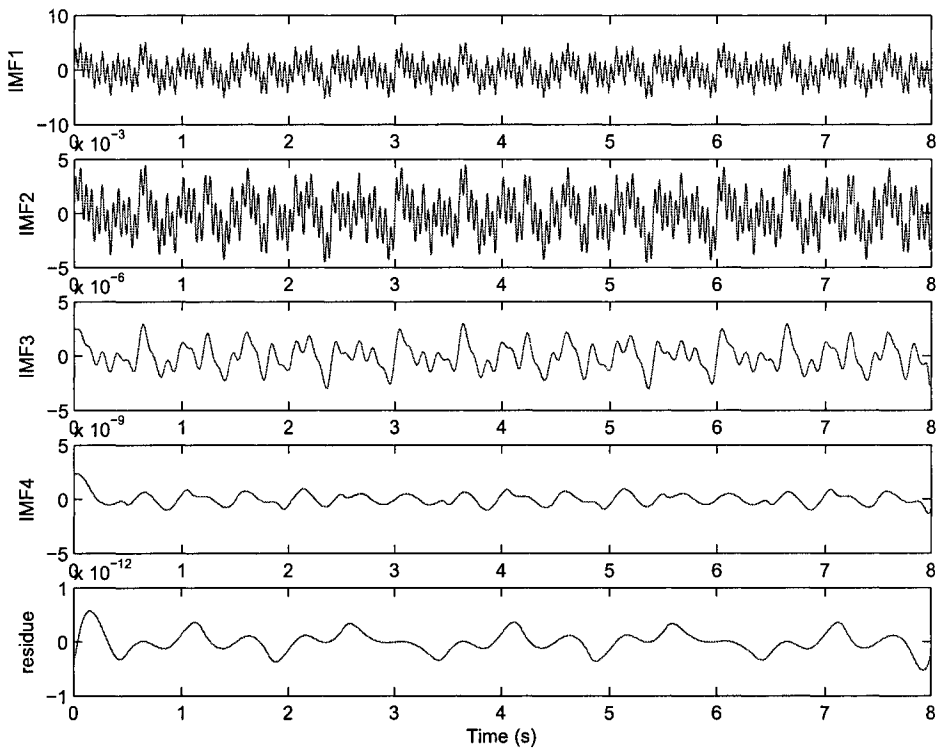

Figure 4.9: Decomposition of an 8000-point sinusoid combination using LMMD.

position capability test because it uses only one average value to represent points between two extrema and it ignores local features too much. The proposed approach using the windowed local mean shows more useful information than does LMMD method; it identifies impulses that hide in chirp signals without losing its basic decomposition capability. This is due to the property of the windowed local mean takes every data point into consideration and captures more local features than does LMMD. Although the LMMD is faster, it sacrifices the basic capability of decomposition and has little to contribute to impulse detection. The proposed approach may have the potential to detect real impulses mixed with other types of background signals or noises.

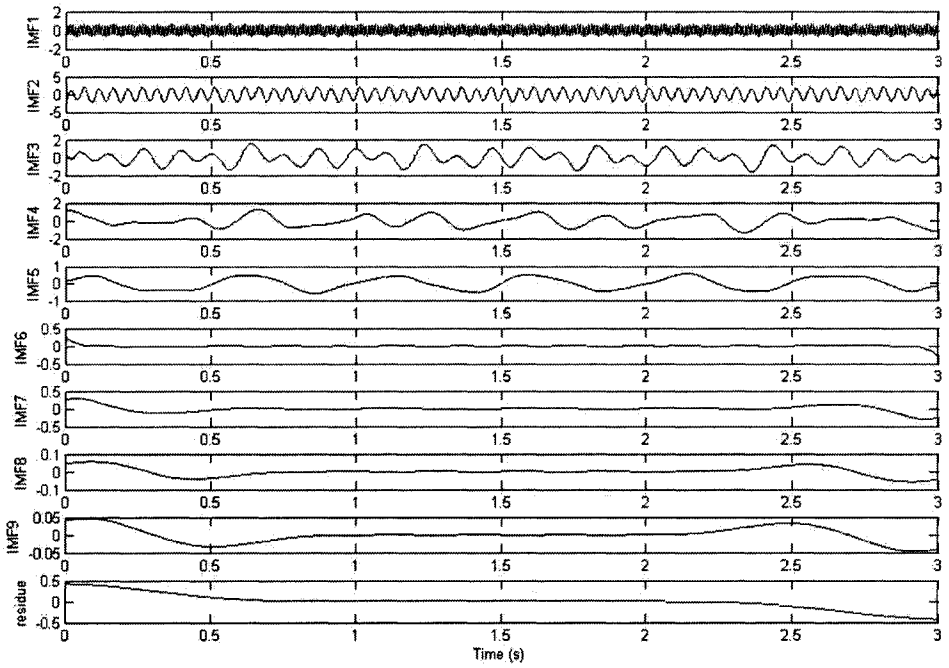

Figure 4.10: Decomposition of a 3000-point sinusoid combination using windowed local mean.

Table 4.3: Comparison of the CPU times for the windowed local mean approximation and the LMMD method on a 3000-point combination of chirp and impulses.

Direct-mean approximation	LMMD	WINDOWED
CPU time	0.1090s	4.4690s

4.4 A Test on Experimental Data

We have verified that the windowed local mean approximation is the winner of the comparisons on simulated signals. Now we look at how it performs on an experimental data. It has been applied to the same experimental data in Chapter 3 and the decomposition result is shown in Fig. 4.18. The windowed local mean approximation takes 13.733 seconds. The first IMF obtained enlarged in Fig. 4.19. We can see that the obtained first IMF also shows impulses

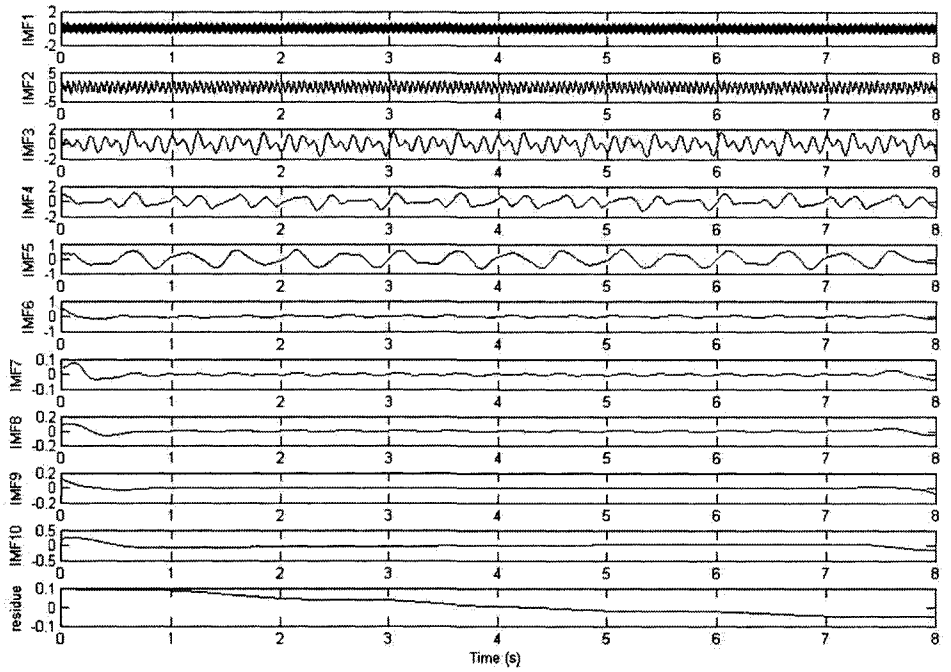

Figure 4.11: Decomposition of an 8000-point sinusoid combination using windowed local mean.

Table 4.4: Comparison of the CPU times for the windowed local mean approximation and the LMMD method on an 8000-point combination of chirp and impulses.

Direct-mean approximation	LMMD	WINDOWED
CPU time	0.6870s	13.3430s

much more clearly than the original data. It is easy to measure that distance between two impulses which is 0.18 seconds, and represents the frequency of the output shaft ($1/0.18 = 5.5$ Hz). Compared with the experimental result by the MMPHI method in Chapter 3, it is hard to give a numerical indicator telling which is better, Fig. 3.16 or Fig. 4.19, but, visually, more of the impulses in the original data can be seen clearly in Fig. 4.19.

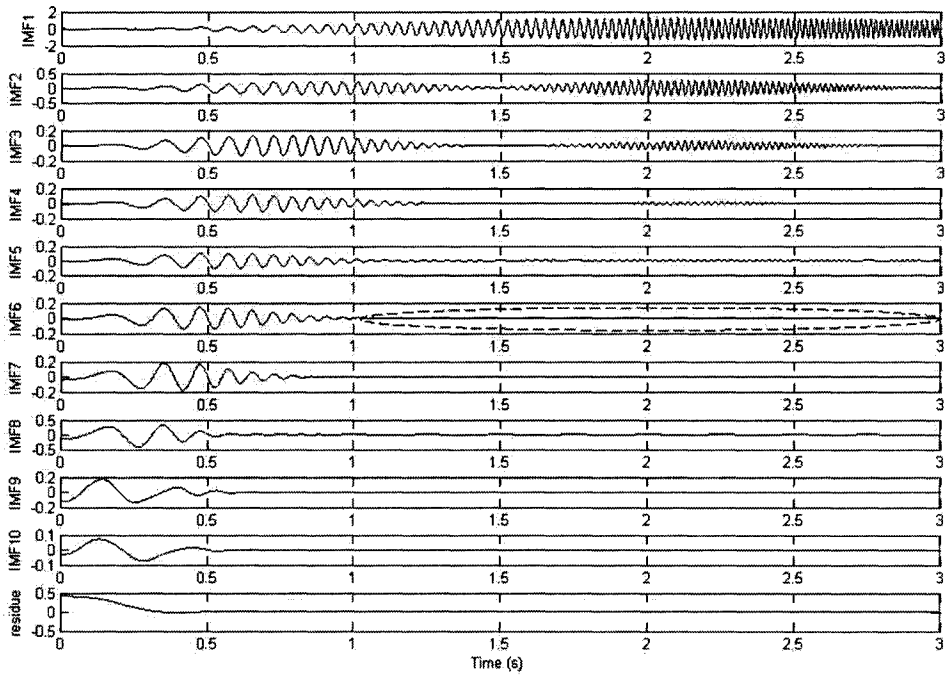

Figure 4.12: Decomposition of a 3000-point combination of impulse and chirp using the windowed local mean.

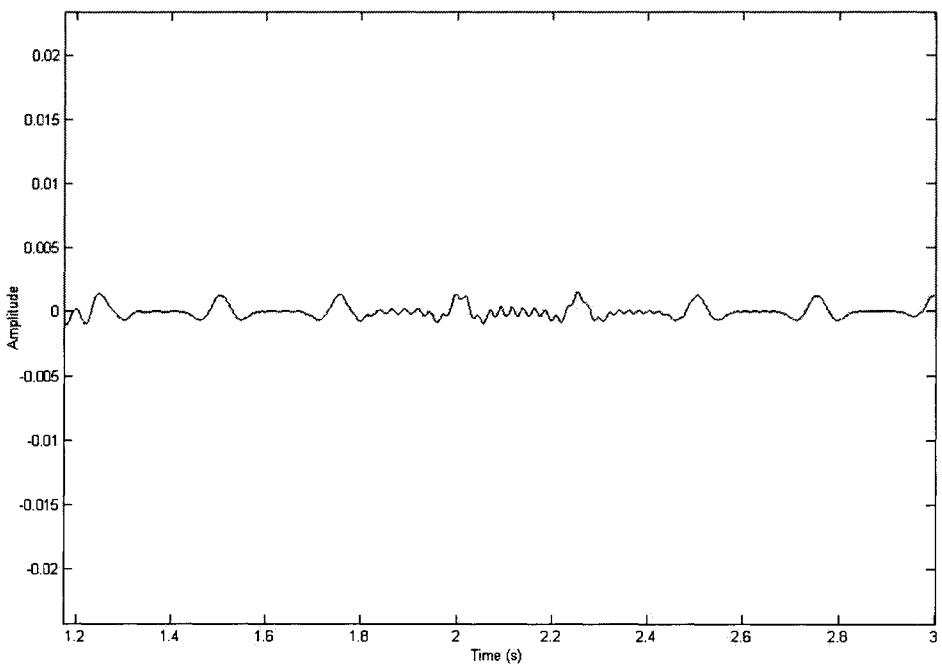

Figure 4.13: The enlarged region as marked in IMF6 of Fig. 4.12.

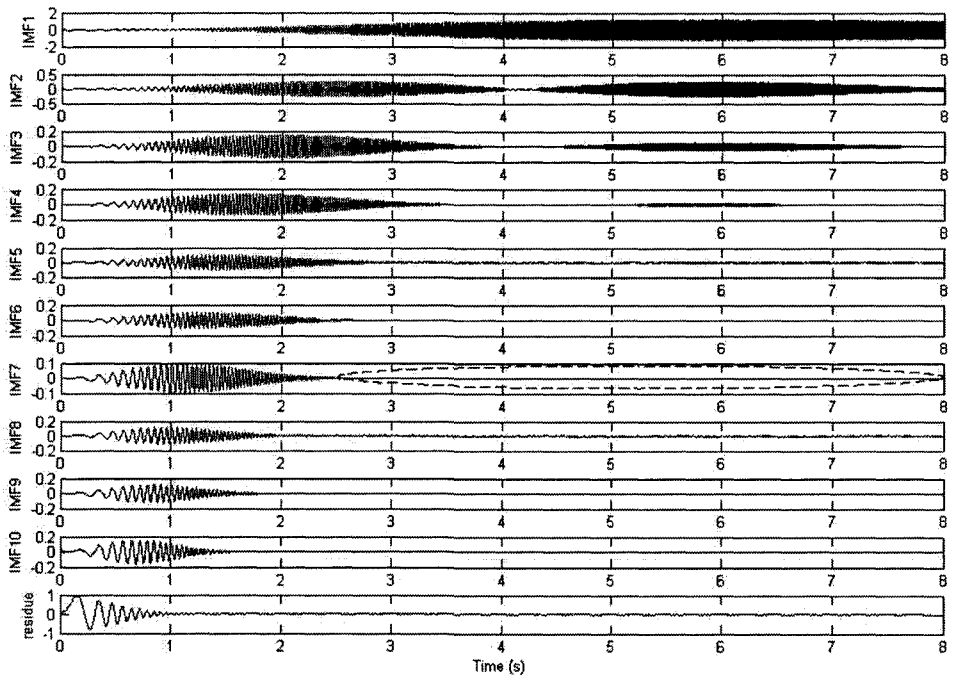

Figure 4.14: Decomposition of an 8000-point combination of impulse and chirp using the windowed local mean.

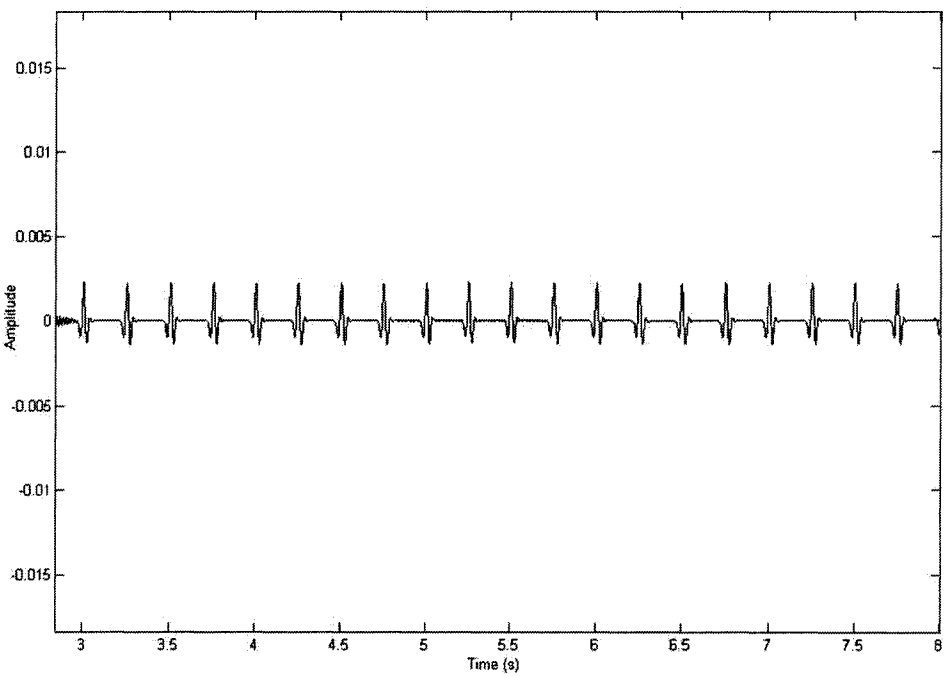

Figure 4.15: The enlarged region as marked in IMF7 of Fig. 4.14.

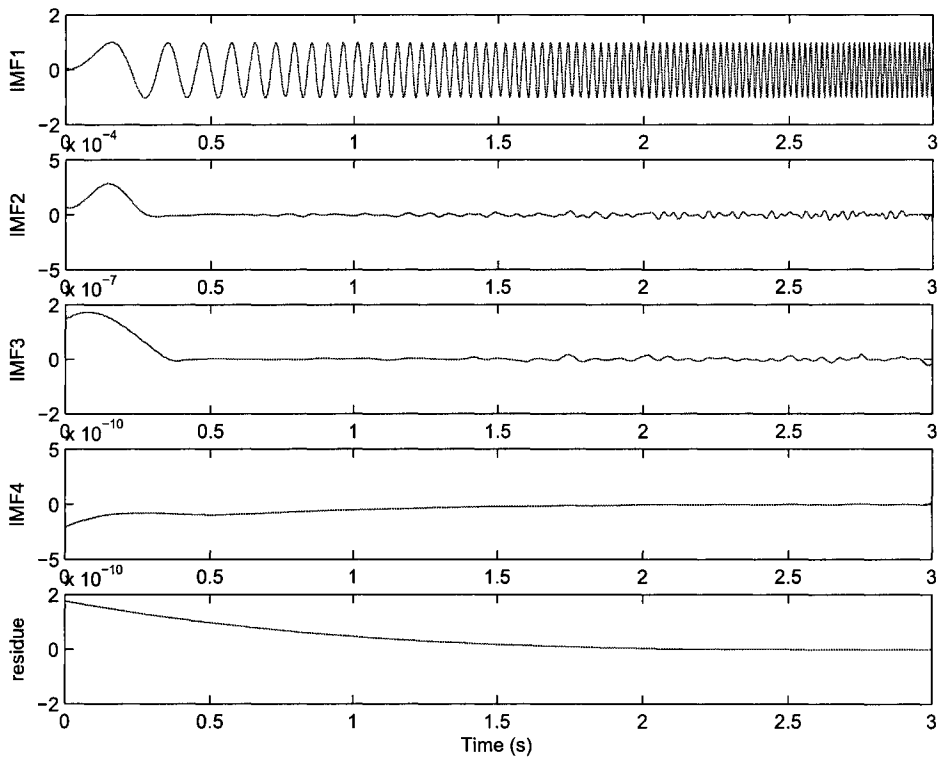

Figure 4.16: Decomposition of a 3000-point combination of impulse and chirp using LMMD.

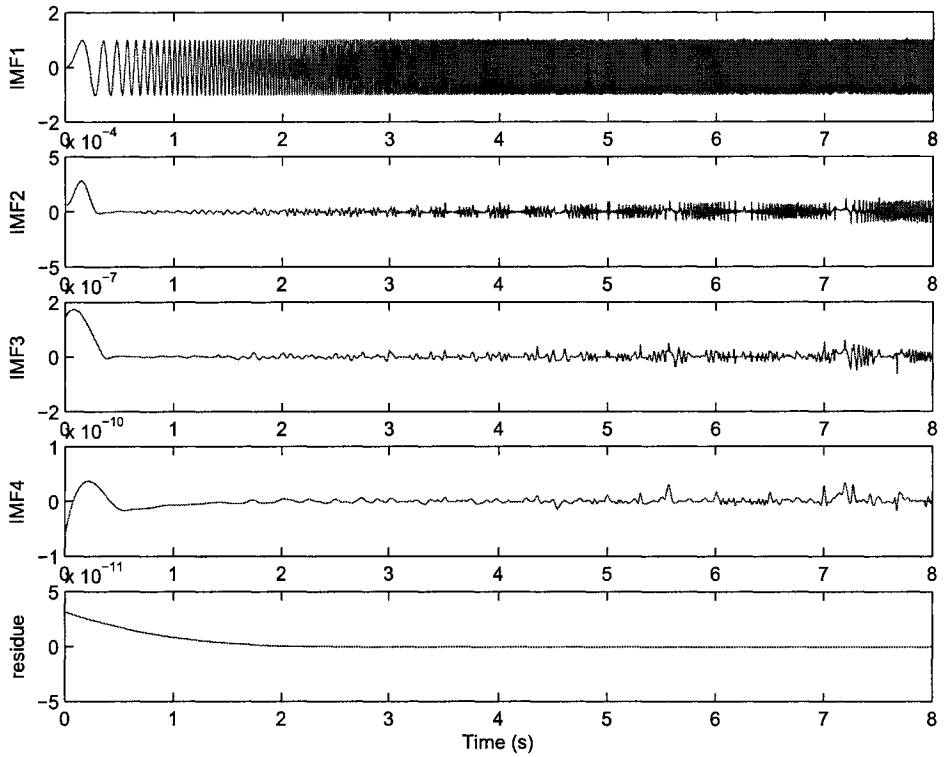

Figure 4.17: Decomposition of an 8000-point combination of impulse and chirp using LMMD.

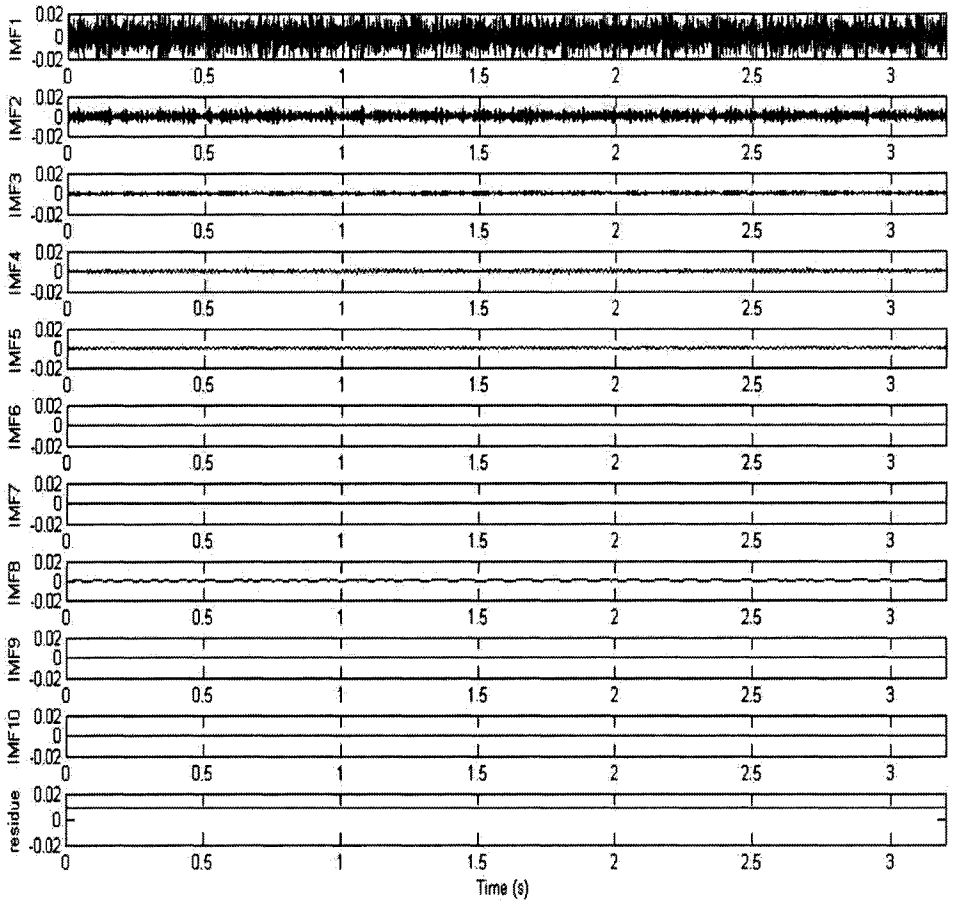

Figure 4.18: Decomposition of the vibration data set using EMD with the windowed local mean approximation.

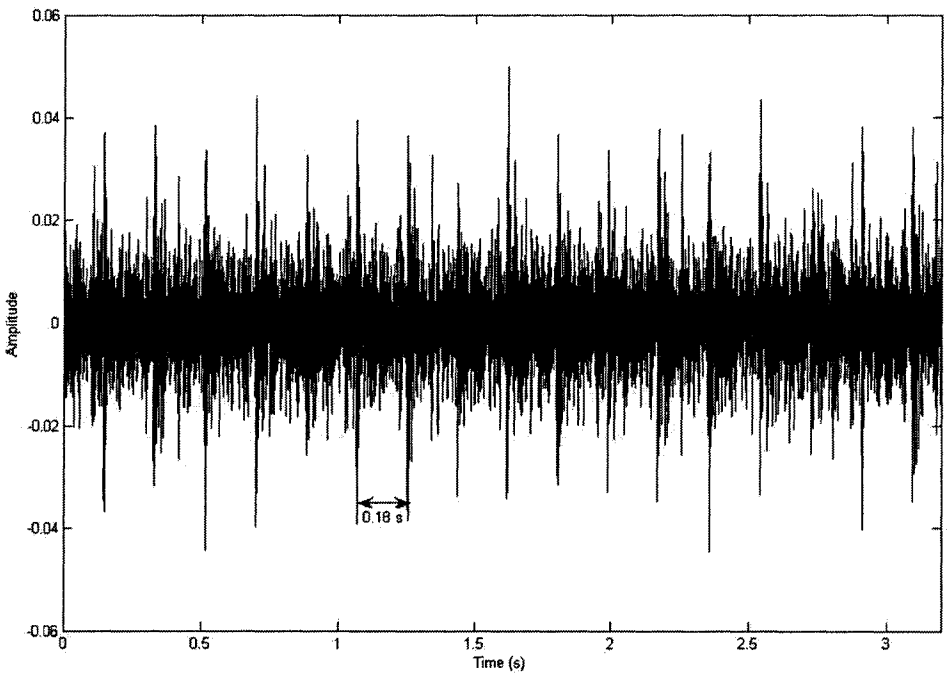

Figure 4.19: The first IMF in Fig. 4.18.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

Chapters 3 and 4 have discussed improvements to the two types of approximation to the local mean in the process of EMD, i.e. the envelope-mean approximation and the direct-mean approximation. Based on the work in this thesis, the following conclusions have been reached:

1. The MMPHI approach has advantages over the other two envelope-mean methods, i.e. the cubic spline and high-order spline methods, with regard to CPU time and accuracy. The high-order spline method consumes too much time and does not gain a benefit on accuracy. The cubic spline method performs very much like MMPHI when testing simulated long impulses and chirp signal. This accuracy assessment is primarily based on visual observations. When a visual observation is not able to detect much difference, the averaged MSE is used to help with the performance assessment.

2. The windowed local mean approximation is better than another direction-mean method, i.e. LMMD method, with regard to its capability to iden-

tify impulses that hide in other signals. This merit can be obtained without sacrificing too much CPU time and without losing the basic decomposition ability. Their difference is obvious through visual observations.

3. When applied to an experimental data, both of improved methods of the two types of approximations help EMD decompose the original data into more useful IMFs. Between them, however, there is not much difference in CPU time. Their capability to identify impulses is hard to define but, from visual observations, the windowed local mean approximation seems to perform better than does MMPHI.

4. In applying EMD to processing signals, we suggest using the MMPHI approach to have a quick look at what basic frequency components are contained in the raw data. When that has been done the windowed local mean approximation can be used to detect impulses and any characteristic frequency that may exist due to faulty conditions in a system being monitored.

5.2 Future Work

The MMPHI approach has been developed to address concerns regarding monotonicity. Monotonicity is not the only property that is important to a curve. For example, convexity is another property that requires the differentiation of the consecutive data points be monotonic. Brodlie and Butt [11] developed a type of piecewise cubic interpolation that preserves convexity. The application of such a type of interpolation to the local mean approximation may improve the accuracy of EMD but more resources and time would be

consumed.

In the window local mean approximation, selection of a proper window width is crucial to the effect of a decomposition. We use a single average width value in each decomposition iteration. The average width may not work well for data without uniform frequencies so that we could use varying widths along the length of the data to capture the local features more precisely.

Last but not least, visual observation is not enough to assess how well impulse identification is being performed. An reasonable indicator of the accuracy of a decomposition needs to be defined, especially when real signals are analyzed.

BIBLIOGRAPHY

[1] F. S. Acton. *Numerical Methods That Work.* Math. Assoc. Amer., 2nd printing, Washington, DC: pp 331-334, 1990.

[2] N. Aretakis and K. Mathioudakis. Classification of radial compressor faults using pattern-recognition techniques. *Control Engineering Practice*, 6:1217–1223, 1998.

[3] M. El Badaoui, V. Cahouet, F. Guillet, J. Daniee Re, and P. Velex. Modeling and detection of localized tooth defects in geared systems. *Transactions of the ASME*, 123:422–430, 2001.

[4] D. C. Baillie and J. Mathew. A comparison of autoregressive modeling techniques for fault diagnosis of rolling element bearings. *Mechanical Systems and Signal Processing*, 10:1–17, 1996.

[5] R. H. Bartels, J. C. Beatty, and B. A. Barsky. *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling.* Morgan Kaufmann Publisher Inc., The United States of America, 1987.

[6] N. Baydar and A. Ball. Detection of gear failures via vibration and acoustic signals using wavelet transform. *Mechanical Systems and Signal Processing*, 17(4):787–804, 2003.

[7] N. Baydar, Q. Chen, A. Ball, and U. Kruger. Detection of incipient tooth defect in helical gears using multivariate statistics. *Mechanical Systems and Signal Processing*, 15:303–321, 2001.

[8] S. Bitterlich, W. Durner, S. C. Iden, and P. Knabner. Inverse estimation of the unsaturated soil hydraulic properties from column outflow experiments using free-form parameterizations. *Vadose Zone Journal*, 3(3):971–981, 2004.

[9] T. Bose. *Digital Signal and Image Processing.* John Wiley & Sons, Inc., Hoboken, New York, 2004.

[10] George E. P. Box and F. M. Jenkins. *Time Series Analysis: Forecasting and Control.* Holden-Day, Oakland, CA, second edition, 1978.

[11] K. W. Brodlie and S. Butt. Preserving convexity using piecewise cubic interpolation. *Computer and Graphics*, 15(1):15–23, 1991.

[12] J. Butland. A method for interpolating reasonable-shaped curves through any data. *Proceedings of Computer Graphics 80*, Online Publications Ltd., Northwood Hills, Middleses, UK:409–422, 1980.

[13] J Charley, G Bodoville, and G Degallaix. Analysis of braking noise and vibration measurements by time-frequency approaches. *Proceedings of The Institution of Mechanical Engineers*, 215, Part C:1381–1400, 2001.

[14] L. Cohen. Time-frequency distribution–a review. *Proceedings of the IEEE*, 77:941–981, 1989.

[15] L. Cohen. *Time-frequency analysis.* Prentice-Hall, Englewood Cliffs, NJ, 1995.

[16] G. Dalpiaz, A. Rivola, and R. Rubini. Effectiveness and sensitivity of vibration processing techniques for local fault detection in gears. *Mechanical Systems and Signal Processing*, 14:387–412, 2000.

[17] J. P. Dron, L. Rasolofondraibe, C. Couet, and A. Pavan. Fault detection and monitoring of a ball bearing benchtest and a production machine via autoregressive spectrum analysis. *Journal of Sound and Vibration*, 218:501–525, 1998.

[18] D. F. Fink and H. W. Beaty. *Standard Handbook for Electrical Engineers.* McGraw-Hill, New York, 1978.

[19] F. N. Fritsch and J. Butland. A method for constructing local monotone piecewise cubic interpolants. *SIAM J. Sci. Stat. Comput.*, 5(2):300–304, 1984.

[20] F. N. Fritsch and R. E. Carlso. Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17(2):238–246, 1980.

[21] Q. Gai, X. J. Ma, H. Y. Zhang, and Y K Zou. Processing time-varying signals by a new method. *Radar, 2001 CIE International Conference on, Proceedings*, pages 15–18, 2001.

[22] S. Goumas, M. Zervakis, A. Pouliezos, and G. S. Stavrakakis. Intelligent on-line quality control of washing machines using discrete wavelet analysis features and likelihood classification. *Engineering Applications of Artificial Intelligence*, 14(5):655–666, 2001.

[23] S. L. Hahn. *Hilbert Transform in Signal Processing.* Artech House, London, 1996.

[24] C. M. Harris and A. G. Piersol. *Harris' Shock and Vibration Handbook.* McGraw-Hill, 2002.

[25] D. J. Huang, J. P. Zhao, and J. L. Su. Practical implementaion of Hilbert-Huang transform algorithm. *Acta Oceanologica Sinica*, 22(1):1–14, 2003.

[26] N. E. Huang, Z. Shen, and S. R. Long. A new view of nonlinear water waves: the Hilbert spectrum. *A. Rev. Fluid Mech.*, 31:417–457, 1999.

[27] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N. C. Yen, C. C. Tung, and H. H. Liu. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Pro. Royal Society*, 454:903–995, 1998.

[28] I. Ihm, D. Cha, and B. Kang. Controllable local monotonic cubic interpolation in fluid animations. *Computer Animation and Vitural Worlds*, 16:365–375, 2005.

[29] A. K. S. Jardine, D. Lin, and D. Banjevic. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Technical Report*, CBM Lab, Dept. of Mechanical and Industrial Engineering, University of Toronto, 2005.

[30] D. Kahaner, C. Moler, and S. Nash. *Numerical Methods and Software.* Prentice Hall, Englewood Cliffs, NJ, 1989.

[31] C. J. Kim and B. D. Russell. Analysis of distribution disturbances and arcing faults using the crest factor. *Electrical Power System Research*, 35:141–148, 1995.

[32] R. W. Komm, F. Hill, and R. Hone. Empirical mode decomposition and Hilbert analysis applied to rotation residuals of the solar convection zone. *The Astrophysical Journal*, 558:428–441, 2001.

[33] J. Lee, R. Abujamra, A. K. S. Jardine, D. Lin, and D. Banjevic. An integrated platform for diagnostics, prognostics and maintenance optimization. *The IMS'2004 International Conference on Advances in Maintenance and in Modeling, Simulation and Intelligent Monitoring of Degradations*, Arles, France, 2005.

[34] J. Levesley, I. J. Anderson, and J. C. Mason (eds.). *Algorithms for Approximation IV, page:24-35.* University of Huddersfield, 2002.

[35] D. Lin, M. Wiseman, D. Banjevic, and A. K. S. Jardine. An approach to signal processing and condition-based maintenance for gearboxes subject to tooth failure. *Mechanical Systems and Signal Processing*, 18(5):993–1007, 2004.

[36] J. Lin and M. J. Zuo. Gearbox fault diagnosis using adaptive wavelet filter. *Mechanical Systems and Signal Processing*, 17(6):1259–1269, 2003.

[37] S.J. Loutridis. Damage detection in gear systems using empirical mode decomposition. *Engineering Structures*, 26:1833–1841, 2004.

[38] K. F. Martin. A review by discussion of condition monitoring and fault-diagnosis in machine-tools. *International Journal of Machine Tools and Manufacture*, 34:527–551, 1994.

[39] A. J. Miller. A new wavelet basis for the decomposition of gear motion error signals and its application to gearbox diagnostics. M.Sc. Thesis,

Graduate Program in Acoustics, The Pennsylvania State University, State College, PA, 1999.

[40] K. Mori, N. Kasashima, T. Yoshioka, and Y. Ueno. Prediction of spalling on a ball bearing by applying the discrete wavelet transform to vibration signals. *Wear*, 195(1-2):162–168, 1996.

[41] N. G. Nikolaou and I. A. Antoniadis. Rolling element bearing fault diagnosis using wavelet packets. *NDT&E International*, 35:197–205, 2001.

[42] A. V. Oppenheim, R. W. Schafter, and J. R. Buck. *Discrete Time Signal Processing*. Prentice Hall, New York, 1999.

[43] C. Pachaud, R. Salvetat, and C. Fray. Crest factor and kurtosis contributions to identify defects inducing periodical impulsive forces. *Mechanical Systems and Signal Processing*, June:903–916, 1997.

[44] J. L. Parrondo, S. Velarde, and C. Santolaria. Development of a predictive maintenance system for a centrifugal pump. *Journal of Quality in Maintenance Engineering*, 4(3):198–211, 1998.

[45] H. Rösler. A study on the empirical mode decomposition. M.Sc. Thesis, Mathematics and Computer Sciences, University of Amsterdam, Amsterdam, Netherlands, 2002.

[46] G. Rilling, P. Flandrin, and P. Goncalves. On empirical mode decomposition and its algorithms. *Proc. IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing NSIP-03*, 2003.

[47] M. J. E. Salami, A. Gani, and T. Pervez. Machine condition monitoring and fault diagnosis using spectral analysis techniques. In *Proceedings*

of the First International Conference on Mechatronics (ICOM '01), volume 2, pages 690–700.

[48] R. G. Stockwell, L. Mansinha, and R. P. Lowe. Time-frequency distribution–a review. *IEEE Transactions on Signal Processing*, 44(4):998–1001, 1996.

[49] C. K. Sung, H. M. Tai, and C. W. Chen. Locating defects of a gear system by the technique of wavelet transform. *Mechanism and Machine Theory*, 35(8):1169–1185, 2000.

[50] X. Tian, J. Lin, M. Agelinchaab, J. Zhang, M. Akhtar, M. J. Zuo, and K R Fyfe. Experiment 02-05 on gearbox. *Technical Report, Department of Mechanical Engineering, University of Alberta, Edmonton, Alberta. 7 pages*, 2002.

[51] P. D. McFadden W. J. Wang. Early detection of gear failure by vibration analysis I. calculation of the time-frequency distribution. *Mechanical Systems and Signal Processing*, 7:193–203, 1993.

[52] C. C. Wang and G. P. J. Too. Rotating machine fault detection based on HOS and artificial neural networks. *Journal of Intelligent Manufacturing*, 13:283–293, 2002.

[53] L. Wang, A. D. Hope, and H. Sadek. Vibration-based condition monitoring of pumps in the waste water industry. *Insight: Non-destructive Testing and Condition Monitoring*, 42(8):500–503, 2000.

[54] W. J. Wang and P. D. McFadden. Decomposition of gear motion signals and its application to gearbox diagnostics. *Journal of Vibration and Acoustics*, 117:363–369, 1995.

[55] W. Q. Wang, F. Ismail, and M. F. Golnarachi. Assessment of gear damage monitoring techniques using vibration measurements. *Mechanical Systems and Signal Processing*, 15(5):905–922, 2001.

[56] Y. Wang, M. J. Zuo, and X. Fan. Design of an experimental system for wear assessment of slurry pumps. *The Second CDEN International Conference on Design Education, Innovation, and Practice*, Kananaskis, Alberta, Canada, 2005.

[57] G. Wolberg and I. Alfy. An energy-minimization framework for monotonic cubic spline interpolation. *Journal of Computational and Applied Mathematics*, 143(2):145–188, 2002.

[58] S. X. Yang, J. S. Hu, Z. T. Wu, and G. B. Yan. Study of empirical decomposition based on high-order spline interpolation. *Journal of Zhejiang University*, 38(3):267–270, 2004.

[59] R. K. Young. *Wavelets Theory and Its Applications.* Kluwer Academic Publishers, Boston, 1993.

[60] D. J. Yu. A fault diagnosis approach for roller bearings based on EMD method and AR model. *Engineering Structures*, 26:1833–1841, 2004.

[61] R. R. Zhang, S. Ma, E. Safak, and S. Hartzell. Hilbert-huang transform analysis of dynamic and earthquake motion recordings. *Journal of Engineering Mechanics*, pages 861–875, 2003.

[62] J. P. Zhao. Improvement of the mirror extending in the empirical mode decomposition method and the technology for eliminating frequency mixing. *High Technology Letters*, 8(3):40–47, 2002.

[63] M. J. Zuo and X. Fan. Empirical mode decomposition and its application to gear fault detection. *Journal of Sound and Vibration*, Revised July 25, 2005.